

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **VI.1 Kesimpulan**

Berdasarkan dari pembahasan pada bab-bab sebelumnya, maka dapat diambil kesimpulan yaitu:

1. Perangkat lunak MSBS atau Music Studio Booking System ini telah berhasil dibangun.
2. Perangkat lunak MSBS atau Music Studio Booking System dapat memberikan kemudahan dalam melakukan reservasi atau booking ruang studio musik serta dapat memberikan informasi kepada pelanggan mengenai ruang studio yang sudah dipesan maupun yang belum dipesan. Perangkat lunak ini juga dapat memberi informasi mengenai tarif ruang studio berdasarkan durasi dan jenis studio yang akan dipesan secara online melalui internet agar mudah diakses dari mana saja.

#### **VI.2 Saran**

Saran dan perbaikan dari pembangunan perangkat lunak Tugas Akhir ini antara lain:

1. Mengoptimalkan fungsi ajax yang telah disediakan oleh Microsoft visual studio agar web yang dibangun terlihat lebih interaktif.
2. Perangkat lunak ini akan lebih baik lagi jika dibangun tidak hanya berbasis web saja, tetapi juga berbasis mobile sehingga kelak aplikasi ini akan semakin mudah diakses dari mana dan kapan saja.

## DAFTAR PUSTAKA

- Anckar, Bill, 2001, "Overcoming Online Booking Barriers With a Software Agent Approach", Global Co-Operation in the New Millennium The 9<sup>th</sup> European Conference on Information Systems, Slovenia.
- Aditomo, Ertanto, 2011, "Aplikasi Pemesanan Alat-Alat Musik Berbasis Web Pada Sambi Studio menggunakan PHP dan MySQL", Fakultas Ilmu Komputer, Program Studi Sistem Informasi, Universitas Mercu Buana, Jakarta.
- Elidjen, Firlyanti, Judodihardjo, S., & Indrajaja, Topik, 2005, "Aplikasi Sistem Penjualan Tiket Bioskop 21 Berbasis Web dan WAP", Seminar Nasional Aplikasi Teknologi Informasi (SNATI 2005), Universitas Bina Nusantara, ISBN: 979-756-061-6.
- Hartono, Jogiyanto, 2005, "Analisis & Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis".
- Honggowibowo, Sediartie, 2005, "Sistem Reservasi Pesawat Terbang Berbasis Web", Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005), Jurusan Teknik Informatika, Sekolah Tinggi Teknologi Adisutjipto, ISBN: 979-756-061-6.
- Hananto, Yudi, 2011, "Pembuatan Website Sebagai Media Promosi dan Reservasi Pada Hotel Bugis Asri Yogyakarta", Jurusan Sistem Informasi, Sekolah Tinggi Manajemen Informatika dan Komputer AMIKOM, Yogyakarta.

- Imam, 2005, "SQL Server 2000", Penerbit Graha Ilmu, Jogja.
- Irianto, Agus, 2009, "Managing Airline Reservation System", Rajawali Pers, Jakarta.
- Järveläinen, Jonna, 2003, "Preferring Offline Bookings: An Empirical Study of Channel Choice Motives of Online Information Seekers", Turku Centre for Computer Science, Turku School of Economics and Business Administration, Finland.
- Korenke, David M., 2004, "Database processing: Fundamental, Design, and Implementation", Pearson Prentice Hall, United States of America.
- Lucas, Henry C., 1981, "The Analysis, Design, and Implementation of Information System", McGraww Hill, Tokyo.
- Ramakrishnan, Gehrke, 2003, "Database Management Systems", McGraww Hill, Singapore.
- Rifai, F.Indra, 2010, "APLIKASI SISTEM INFORMASI KERETA API DAN PEMESANAN TIKET BERBASIS J2ME DI DAERAH OPERASI VI", Jurusan Teknik Elektro Fakultas Teknik, Universitas Muhammadiyah Surakarta, Surakarta.
- Setiawan, Anton H & Sediartie, Titien, 2005, "SISTEM RESERVASI PESAWAT TERBANG BERBASIS WEB", *Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005)*, Sekolah Tinggi Teknologi Adisutjipto, ISBN: 979-756-061-6.
- Setiawan, Elisabet, 2007, "Aplikasi Komputer Berbasis Web untuk Menangani Reservasi, Check-In, dan Check-Out Secara Online dengan Studi Kasus

"Hotel Cipaku Indah" - Bandung", Jurnal Sistem Informasi ,Vol. 2, No. 1, pp.51-68.

Tiasaputri, Nurma, 2010, "MANFAAT *GABRIELLE SYSTEM* PADA RESERVASI DAN TIKETING SRIWIJAYA AIR DISTRIK SOLO", Fakultas Sastra dan Seni Rupa, Universitas Sebelas Maret, Surakarta.

Yusrizal, Mochamad, 2011, "Aplikasi Booking Online di Studio Masterplan Recording Chamber", Jurusan Teknik Informatika, Universitas Komputer Indonesia, Bandung.



# SKPL

## SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

### MSBS

(Music Studio Booking System)

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Bagus Perwira Laksmana / 5177

Program Studi Teknik Informatika – Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		<b>SKPL-MSBS</b>		1/39
		Revisi		

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

## Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi
			

## Daftar Isi

1	Pendahuluan.....	6
1.1	Tujuan.....	6
1.2	Lingkup Masalah.....	6
1.3	Definisi, Akronim dan Singkatan.....	7
	Server.....	7
	Database.....	7
	DBMS.....	7
	GUI.....	7
1.4	Referensi.....	8
1.5	Deskripsi umum (Overview).....	8
2	Deskripsi Kebutuhan.....	9
2.1	Perspektif produk.....	9
2.2	Fungsi Produk.....	10
2.3	Karakteristik Pengguna.....	15
2.4	Batasan-batasan.....	16
2.5	Asumsi dan Ketergantungan.....	16
3	Kebutuhan khusus.....	17
3.1	Kebutuhan antarmuka eksternal.....	17
3.2	Kebutuhan fungsionalitas Perangkat Lunak.....	18
4	Spesifikasi Rinci Kebutuhan.....	19
4.1	Spesifikasi Kebutuhan Fungsionalitas.....	20
5	Entity Relationship Diagram (ERD).....	41



## Daftar Gambar

1. Arsitektur Perangkat Lunak MSBS.....	10
2. Use Case Diagram .....	25
3. Entity Relationship Diagram (ERD) .....	63

## **1 Pendahuluan**

### **1.1 Tujuan**

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak MSBS (Music Studio Booking System) untuk mendefinisikan kebutuhan perangkat lunak, yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna), dan atribut (fitur-fitur tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-MSBS ini juga mendefinisikan batasan perancangan perangkat lunak.

### **1.2 Lingkup Masalah**

Perangkat Lunak MSBS dikembangkan dengan tujuan untuk :

1. Menangani Login.
2. Menangani pengelolaan data pengguna sistem.
3. Menangani pengelolaan data member.
4. Menangani pengelolaan data studio musik.
5. Menangani pengelolaan data booking.
6. Menangani pengelolaan data pertanyaan.
7. Fungsi Generate Laporan Pemesanan Harian.
8. Menangani registrasi member.
9. Menangani pengubahan profil member.
10. Menangani pengiriman pertanyaan, kritik, dan saran dari member EHA Music Studio kepada pihak manajemen.
11. Menangani booking studio secara online.
12. Menangani lihat jadwal studio musik secara online.

Program Studi Teknik Informatika	SKPL – MSBS	6/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

### 1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-MSBS-XX-YY	Kode yang merepresentasikan kebutuhan pada MSBS (Music Studio Booking System). Di mana XX adalah kode bagian, dan YY adalah nomor method produk.
MSBS	Music Studio Booking System. Merupakan perangkat lunak pengelolaan untuk reservasi kamar hotel.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Database	Kumpulan data yang terkait yang diorganisasikan dalam struktur tertentu dan dapat diakses dengan cepat.
DBMS	DataBase Management System atau pengelola manajemen data base.
GUI	GUI adalah <i>Graphical User Interface</i> , yaitu tampilan antarmuka program untuk pengguna

#### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Bagus Perwira Laksmana, *SKPL-AVN*, Universitas Atma Jaya Yogyakarta, 2011.
2. Malindo, Robertus Apri, *Spesifikasi Kebutuhan Perangkat Lunak TA*, Universitas Atma Jaya Yogyakarta, 2011.

#### 1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 6 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak MSBS yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak MSBS tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak MSBS yang akan dikembangkan, mencakup kebutuhan antarmuka eksternal yaitu antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, dan antarmuka komunikasi. Juga kebutuhan fungsionalitas perangkat lunak yaitu digambarkan dalam use case diagram.

Bagian keempat berisi tentang penjelasan rinci tentang kebutuhan perangkat lunak MSBS yang akan dikembangkan.

Bagian kelima berisi tentang Entity Relationship Diagram yang menjadi dasar pengembangan perangkat lunak MSBS.

Pada bagian terakhir yaitu bagian keenam, berisi tentang kamus data yang menjadi panduan tentang keseluruhan isi SKPL ini.

## **2 Deskripsi Kebutuhan**

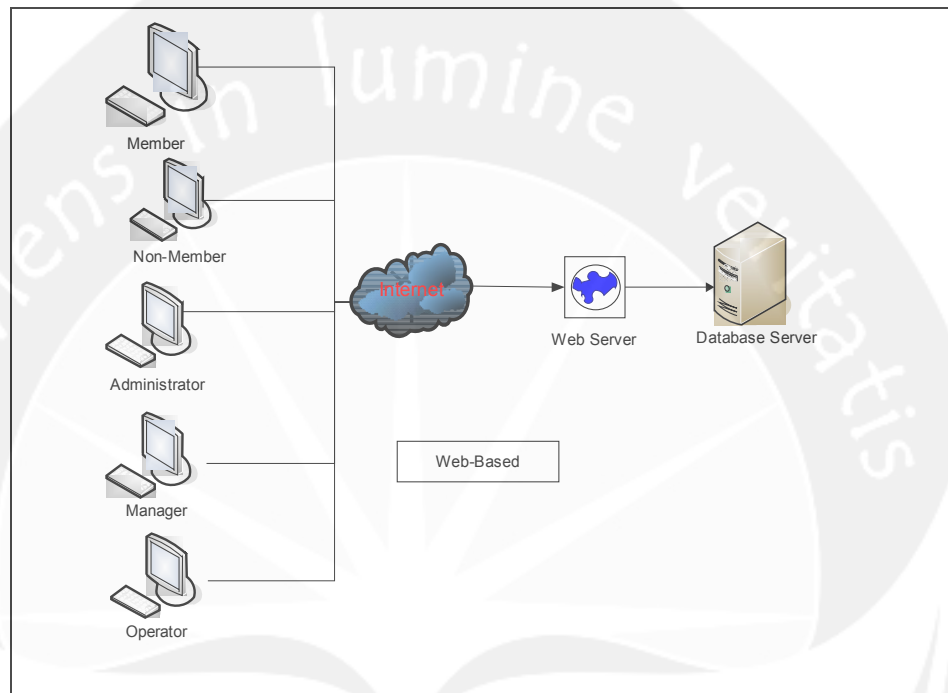
### **2.1 Perspektif produk**

MSBS merupakan perangkat lunak yang dikembangkan untuk membantu pelayanan dan pengelolaan untuk melakukan pemesanan studio musik oleh member via *internet (website)*. Sistem ini menangani penyimpanan data pemesanan studio yang dilakukan sendiri oleh calon member dan juga menangani penyimpanan data calon member itu sendiri.

Pemesanan studio musik ini berjalan pada platform Windows dan dibuat dengan menggunakan bahasa pemrograman Microsoft Visual C#. Sedangkan untuk lingkungan pemrogramannya menggunakan Microsoft Visual Studio 2005.

Pengguna sistem akan berinteraksi dengan sistem melalui antarmuka GUI (Graphical User Interface). Di sini, untuk melakukan interaksi pada aplikasi *web-based*, pengguna sistem harus terhubung dengan internet

yang akan mengakses web server untuk dapat mengirim data yang di-request pengguna sistem.



Gambar 1. Arsitektur Perangkat Lunak MSBS

## 2.2 Fungsi Produk

Fungsi produk perangkat lunak MSBS yaitu :

### 1. Fungsi Login (SKPL-MSBS-01).

Merupakan fungsi yang digunakan oleh pengguna sistem (administrator, manager, operator, dan member) untuk dapat masuk dalam sistem.

### 2. Fungsi Pengelolaan Data Pengguna (SKPL-MSBS-02).

Merupakan fungsi yang digunakan oleh administrator untuk mengelola data pengguna sistem yang

berinteraksi dengan sistem (administrator, manager, operator, member).

Fungsi pengelolaan data pengguna mencakup:

**a. Fungsi Tambah Pengguna (SKPL-MSBS-02-01).**

Merupakan fungsi untuk menambahkan data pengguna sistem yang baru.

**b. Fungsi Edit Pengguna (SKPL-MSBS-02-02).**

Merupakan fungsi untuk mengubah data pengguna sistem jika ada perubahan data pengguna sistem lama.

**c. Fungsi Hapus Pengguna (SKPL-MSBS-02-03).**

Merupakan fungsi untuk menghapus data pengguna sistem yang tidak diperlukan.

**d. Fungsi Tampil Pengguna (SKPL-MSBS-02-04).**

Merupakan fungsi untuk menampilkan data pengguna sistem.

**e. Fungsi Cari Pengguna (SKPL-OSA-02-05).**

Merupakan fungsi untuk mencari data-data user berdasarkan kata kunci yang diinputkan.

**3. Fungsi Pengelolaan Data Member (SKPL- MSBS-03).**

Merupakan fungsi yang digunakan oleh administrator untuk mengelola data member.

Fungsi pengelolaan data member mencakup :

**a. Fungsi Hapus Member (SKPL-MSBS-03-01).**

Merupakan fungsi untuk menghapus data member yang tidak diperlukan.

**b. Fungsi Tampil Member (SKPL- MSBS -03-02).**

Merupakan fungsi untuk menampilkan data member.

**c. Fungsi Cari Member (SKPL- MSBS -03-03) .**

Merupakan fungsi untuk mencari data-data member berdasarkan kata kunci yang diinputkan.

**4. Fungsi Pengelolaan Data Studio Musik (SKPL-MSBS-04)**

Merupakan fungsi yang digunakan oleh manager untuk mengelola data studio musik. Fungsi di dalamnya meliputi:

**a.Fungsi Entry Data Studio Musik (SKPL-MSBS-04-01) .**

Merupakan fungsi yang digunakan untuk menambahkan data studio musik.

**b.Fungsi Edit Data Studio Musik (SKPL-MSBS-04-02) .**

Merupakan fungsi yang digunakan untuk mengubah data studio musik.

**c.Fungsi Display Data Studio Musik (SKPL-MSBS-04-03) .**

Merupakan fungsi yang digunakan untuk menampilkan data studio musik.

**d. Fungsi Search Data Studio Musik (SKPL-MSBS-04-04) .**

Merupakan fungsi yang digunakan untuk mencari data studio musik.

**e. Fungsi Delete Data Studio Musik (SKPL-MSBS-04-05)**



Merupakan fungsi yang digunakan untuk menghapus data studio musik.

#### **5. Fungsi Pengelolaan Data Booking (SKPL-MSBS-05)**

Merupakan fungsi yang digunakan oleh operator untuk mengelola data booking. Fungsi di dalamnya meliputi:

- a. Fungsi Pembatalan Booking (SKPL-MSBS-05-01).** Merupakan fungsi yang digunakan untuk membatalkan data booking yang tidak diperlukan.
- b. Fungsi Tampil Booking (SKPL-MSBS-05-02).** Merupakan fungsi yang digunakan untuk menampilkan data booking.
- c. Fungsi Cari Data Booking (SKPL-MSBS-05-03).**  
Merupakan fungsi yang digunakan untuk mencari data booking.

#### **6. Fungsi Pengelolaan Data Pertanyaan (SKPL-MSBS-06).**

Merupakan fungsi yang digunakan oleh operator untuk mengelola data pertanyaan. Fungsi di dalamnya meliputi:

- a. Fungsi Jawab Data Pertanyaan (SKPL-MSBS-06-01).**

Merupakan fungsi yang digunakan untuk menjawab data pertanyaan.

**b. Fungsi Tampil Data Pertanyaan (SKPL-MSBS-06-02) .**

Merupakan fungsi yang digunakan untuk menampilkan data pertanyaan.

**c. Fungsi Cari Data Pertanyaan (SKPL-MSBS-06-03) .**

Merupakan fungsi yang digunakan untuk mencari data pertanyaan.

**7. Fungsi Generate Laporan Pemesanan Harian (SKPL-MSBS-07) .**

Merupakan fungsi yang digunakan oleh manager untuk melakukan generate laporan pemesanan harian.

**8. Fungsi Registrasi Member (SKPL-MSBS-08) .**

Merupakan fungsi yang digunakan oleh nonmember untuk melakukan registrasi atau pendaftaran menjadi member.

**9. Fungsi Ubah Profil Member (SKPL-MSBS-09) .**

Merupakan fungsi yang digunakan oleh member untuk mengubah profil member yang dimiliki.

**10. Fungsi Pengiriman Pertanyaan (SKPL-MSBS-10) .**

Merupakan fungsi yang digunakan oleh member maupun nonmember untuk mengirimkan pertanyaan, kritik, maupun saran kepada pihak manajemen EHA Music Studio.

**11. Fungsi Booking Studio (SKPL-MSBS-11) .**

Merupakan fungsi yang digunakan oleh member untuk melakukan pemesanan studio secara online.

#### **12. Fungsi Lihat Jadwal Studio Musik (SKPL-MSBS-12)**

Merupakan fungsi yang digunakan oleh non-member maupun member untuk melihat jadwal studio musik yang sudah dipesan maupun yang belum dipesan.

### **2.3 Karakteristik Pengguna**

Pengguna dari perangkat lunak MSBS ini, yaitu Administrator, Manager, Member, dan Nonmember. Karakteristik yang dimiliki setiap pengguna adalah sebagai berikut:

#### **1. Administrator**

Karakteristik dari pengguna dengan role administrator adalah:

- a. Memahami pengoperasian komputer secara aktif.
- b. Memahami sistem komputer tempat perangkat lunak dijalankan.
- c. Mengerti dan memahami perangkat lunak yang digunakan.

#### **2. Manager**

Karakteristik dari pengguna dengan role manager adalah:

- a. Memahami pengoperasian komputer secara aktif.
- b. Mengerti dan memahami perangkat lunak yang digunakan.

#### **3. Member**

Karakteristik dari pengguna dengan role member adalah:

- a. Mengerti pengoperasian komputer.
  - b. Mengerti penggunaan internet.
4. Nonmember
- a. Mengerti pengoperasian komputer.
  - b. Mengerti penggunaan internet.

#### **2.4 Batasan-batasan**

Batasan-batasan dalam pengembangan perangkat lunak MSBS tersebut adalah :

1. Kebijaksanaan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak MSBS.

2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

#### **2.5 Asumsi dan Ketergantungan**

Perangkan lunak MSBS dikembangkan sebagai aplikasi berbasis website yang dijalankan pada perangkat komputer non mobile, seperti PC desktop dan laptop. Sistem dapat dijalankan pada komputer yang telah memiliki sistem operasi Windows.

### **3 Kebutuhan khusus**

#### **3.1 Kebutuhan antarmuka eksternal**

Kebutuhan antar muka eksternal pada perangkat lunak MSBS meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

##### **3.1.1 Antarmuka pemakai**

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam layar komputer dalam bentuk form-form.

##### **3.1.2 Antarmuka perangkat keras**

Antarmuka perangkat keras yang digunakan dalam perangkat lunak MSBS adalah komputer desktop (PC), laptop, atau notebook.

##### **3.1.3 Antarmuka perangkat lunak**

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak MSBS adalah sebagai berikut :

1. Nama : SQL Server 2005  
Sumber : Microsoft  
Sebagai database management system (DBMS) yang digunakan untuk penyimpanan data di sisi server.
2. Nama : Windows  
Sumber : Microsoft.  
Sebagai sistem operasi computer yang digunakan.
3. Nama : IIS  
Sumber : Microsoft.

Sebagai web server (untuk aplikasi web-based).

4. Nama : .NET Compact Framework 2.0

Sumber : Microsoft.

Sebagai framework untuk menjalankan aplikasi.

5. Nama : Internet Explorer

Sumber : Microsoft.

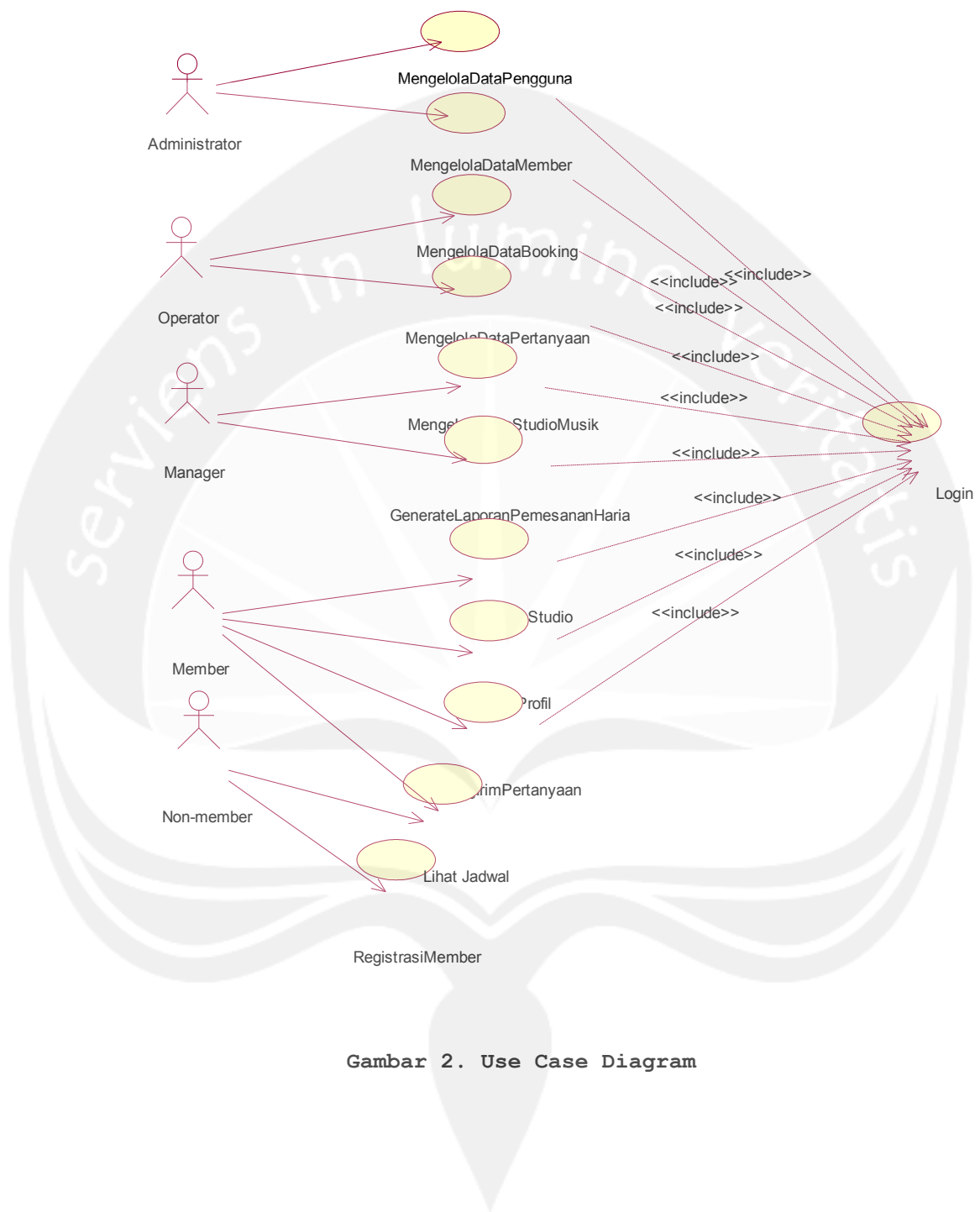
Sebagai browser internet.

#### **3.1.4 Antarmuka Komunikasi**

Antarmuka komunikasi perangkat lunak MSBS menggunakan protocol HTTP.

### **3.2 Kebutuhan fungsionalitas Perangkat Lunak**

#### **3.2.1 Use Case Diagram**



Gambar 2. Use Case Diagram

#### 4 Spesifikasi Rinci Kebutuhan

## 4.1 Spesifikasi Kebutuhan Fungsionalitas

### 4.1.1 Use case Spesification : Login

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik dari user dan password yang berupa rangkaian karakter.

#### 2. Primary Actor

1. User

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login
2. Sistem menampilkan antarmuka untuk login
3. Aktor memasukkan id dan password
4. Sistem memeriksa id dan password yang diinputkan aktor
- E-1 Password atau id user tidak sesuai
5. Sistem memberikan akses ke aktor
6. Use Case ini selesai

#### 5. Alternative Flow

none

#### 6. Error Flow

E-1 Password atau nama user tidak sesuai

1. Sistem menampilkan peringatan bahwa id user atau password tidak sesuai
2. Kembali ke Basic Flow langkah ke 3

#### 7. PreConditions

none

#### 8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

### 4.1.2 Use case Spesification : Pengelolaan Data Pengguna

Program Studi Teknik Informatika	SKPL – MSBS	20/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



## 1. Brief Description

Use Case ini memungkinkan administrator untuk melakukan pengelolaan data pengguna yang berinteraksi dengan sistem, termasuk didalamnya menambah data pengguna, mengedit data pengguna, menghapus data pengguna, menampilkan data pengguna, dan mencari data pengguna.

## 2. Primary Actor

1. Administrator

## 3. Supporting Actor

none

## 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data pengguna.
2. Sistem memberikan pilihan untuk melakukan penambahan data pengguna, pengeditan data pengguna, penghapusan data pengguna, pencarian data pengguna, atau menampilkan data pengguna.
3. Aktor memilih untuk melakukan penambahan data pengguna
  - A-1 Aktor memilih untuk melakukan pengeditan data pengguna
  - A-2 Aktor memilih untuk melakukan penghapusan data pengguna
  - A-3 Aktor memilih untuk menampilkan data pengguna
  - A-4 Aktor memilih untuk melakukan pencarian data pengguna
4. Sistem menampilkan form untuk memasukkan data calon pengguna.
5. Aktor memasukkan data calon pengguna yang akan dijadikan pengguna.
6. Sistem melakukan pengecekan terhadap data calon pengguna yang diinputkan aktor

E-1 Data yang diinputkan salah

Program Studi Teknik Informatika	SKPL – MSBS	21/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-2 Data yang diinputkan sudah ada

7. Aktor meminta sistem untuk menyimpan data yang telah diinputkan
8. Sistem menyimpan data yang telah diinputkan aktor.
9. Use Case selesai

#### 5. Alternative Flow

A-1 Aktor memilih untuk melakukan pengeditan data pengguna

1. Sistem menampilkan form untuk memasukkan data pengguna yang akan diedit.
2. Aktor memasukkan data pengguna yang ingin diedit
3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor

E-3 Data yang diinputkan aktor tidak ditemukan

4. Sistem menampilkan data pengguna yang ditemukan berdasarkan inputan aktor
5. Aktor mengubah data pengguna yang ditampilkan sistem
6. Sistem melakukan pengecekan terhadap data calon pengguna yang diinputkan aktor

E-4 Data yang diinputkan salah

7. Aktor meminta sistem untuk mengupdate data pengguna yang telah diubah
8. Sistem mengupdate data pengguna berdasarkan ubahan aktor
9. Berlanjut ke Basic Flow langkah ke 9

A-2 Aktor memilih untuk melakukan penghapusan data pengguna

1. Sistem menampilkan form untuk memasukkan data pengguna yang akan dihapus
2. Aktor memasukkan data pengguna yang ingin dihapus

3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor

E-5 Data yang diinputkan aktor tidak ditemukan

4. Sistem menampilkan data pengguna yang ditemukan berdasarkan inputan aktor

5. Aktor meminta sistem untuk menghapus data pengguna yang ditampilkan sistem

6. Sistem melakukan penghapusan terhadap data pengguna yang diminta aktor

7. Berlanjut ke Basic Flow langkah ke 9

A-3 Aktor memilih untuk menampilkan data pengguna

1. Sistem menampilkan seluruh data pengguna yang tersimpan

2. Berlanjut ke Basic Flow langkah ke 9

A-4 Aktor memilih untuk melakukan pencarian data pengguna

1. Sistem menampilkan form untuk memasukkan data pengguna yang akan dicari

2. Aktor memasukkan data pengguna yang ingin dicari

3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor

E-6 Data yang diinputkan aktor tidak ditemukan

4. Sistem menampilkan data pengguna yang dicari aktor

5. Berlanjut ke Basic Flow langkah ke 9

## 6. Error Flow

E-1 Data yang diinputkan salah

1. Sistem memberikan peringatan bahwa data yang diinputkan salah

2. Kembali ke Basic Flow langkah ke 5

E-2 Data yang diinputkan sudah ada

1. Sistem memberikan peringatan bahwa data pengguna yang diinputkan telah ada

2. Kembali ke Basic Flow langkah ke 5
- E-3 Data yang diinputkan aktor tidak ditemukan
  1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan
  2. Kembali ke Alternative Flow A-1 langkah ke 2
- E-4 Data yang diinputkan salah
  1. Sistem memberikan peringatan bahwa data yang diinputkan salah
  2. Kembali ke Alternative Flow A-1 langkah ke 5
- E-5 Data yang diinputkan aktor tidak ditemukan
  1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan
  2. Kembali ke Alternative Flow A-2 langkah ke 2
- E-6 Data yang diinputkan aktor tidak ditemukan
  1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan
  2. Kembali ke Alternative Flow A-4 langkah ke 2

#### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

1. Data pengguna di database telah terupdate

### **4.1.3**

#### **Use case Spesification : Pengelolaan Data Member**

##### **1. Brief Description**

Use Case ini memungkinkan administrator untuk melakukan pengelolaan data member, termasuk di dalamnya menghapus data member, menampilkan data member, dan mencari data member.

##### **2. Primary Actor**

1. Administrator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

Program Studi Teknik Informatika	SKPL – MSBS	24/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data member.
2. Sistem memberikan pilihan untuk melakukan penghapusan data member, pencarian data member, atau menampilkan data member.
3. Aktor memilih untuk melakukan penghapusan data member
  - A-1 Aktor memilih untuk menampilkan data member
  - A-2 Aktor memilih untuk melakukan pencarian data member
4. Sistem menampilkan form untuk memasukkan data member yang akan dihapus
5. Aktor memasukkan data member yang ingin dihapus
6. Sistem melakukan pengecekan terhadap data yang diinputkan aktor
  - E-1 Data yang diinputkan aktor tidak ditemukan
7. Sistem menampilkan data member yang ditemukan berdasarkan inputan aktor
8. Aktor meminta sistem untuk menghapus data member yang ditampilkan sistem
9. Sistem melakukan penghapusan terhadap data member yang diminta aktor
10. Use Case selesai

#### 5. Alternative Flow

- A-1 Aktor memilih untuk menampilkan data member
  1. Sistem menampilkan seluruh data member yang tersimpan
  2. Berlanjut ke Basic Flow langkah ke-10
- A-2 Aktor memilih untuk melakukan pencarian data member
  1. Sistem menampilkan form untuk memasukkan data member yang akan dicari
  2. Aktor memasukkan data member yang ingin dicari
  3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor

E-2 Data yang diinputkan aktor tidak ditemukan

4. Sistem menampilkan data member yang dicari aktor

5. Berlanjut ke Basic Flow langkah ke-10

#### **6. Error Flow**

E-1 Data yang diinputkan aktor tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan

2. Kembali ke Basic Flow langkah ke-5

E-2 Data yang diinputkan aktor tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan

2. Kembali ke Alternative Flow A-2 langkah ke-2

#### **7. PreConditions**

1. Use Case Login sudah dilakukan

2. Aktor telah memasuki sistem

#### **8. PostConditions**

1. Data member di database telah terupdate

### **4.1.4**

#### **Use case Spesification : Pengelolaan Data Studio**

Musik

##### **1. Brief Description**

Use Case ini memungkinkan administrator untuk melakukan pengelolaan data studio musik, termasuk didalamnya menambah data studio musik, mengedit data studio musik, dan menampilkan data studio musik.

##### **2. Primary Actor**

1. Administrator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

Program Studi Teknik Informatika	SKPL – MSBS	26/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data studio musik.
2. Sistem memberikan pilihan untuk melakukan penambahan data studio musik, pengeditan data studio musik, atau menampilkan data studio musik.
3. Aktor memilih untuk melakukan penambahan data studio musik
  - A-1 Aktor memilih untuk pengeditan data studio musik
  - A-2 Aktor memilih untuk menampilkan data studio musik
4. Sistem menampilkan form untuk memasukkan data studio music.
5. Aktor memasukkan data studio musik.
6. Sistem melakukan pengecekan terhadap data studio musik yang diinputkan aktor
  - E-1 Data yang diinputkan aktor salah
  - E-2 Data yang diinputkan aktor sudah ada
7. Aktor meminta sistem untuk menyimpan data yang telah diinputkan.
8. Sistem menyimpan data yang telah diinputkan aktor.
9. Use Case selesai.

#### 5. Alternative Flow

- A-1 Aktor memilih untuk menampilkan data studio musik
  1. Sistem menampilkan seluruh data member yang tersimpan
  2. Berlanjut ke Basic Flow langkah ke-10
- A-2 Aktor memilih untuk melakukan pengeditan data studio musik
  1. Sistem menampilkan form untuk memasukkan data studio musik yang akan diedit
  2. Aktor memasukkan data studio musik yang ingin diedit

3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor

E-3 Data yang diinputkan aktor tidak ditemukan

4. Sistem menampilkan data studi musik yang ditemukan berdasarkan inputan aktor.

5. Aktor mengubah data studio music yang ditampilkan sistem.

6. Sistem melakukan pengecekan terhadap perubahan data studio music yang diinputkan aktor

E-4 Data yang diinputkan salah

7. Aktor meminta system untuk mengupdate data studio music yang telah diubah.

8. Sistem mengupdate data studio musik berdasarkan ubahan aktor

9. Berlanjut ke basic flow langkah ke-9.

## 6. Error Flow

E-1 Data yang diinputkan aktor tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan

2. Kembali ke Basic Flow langkah ke-5

E-2 Data yang diinputkan aktor sudah ada

1. Sistem memberikan peringatan bahwa data yang diinputkan sudah ada

2. Kembali ke Basic Flow langkah ke-5

E-3 Data yang diinputkan aktor tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan.

2. Kembali ke Alternatif Flow A-2 langkah ke-2

E-4 Data yang diinputkan aktor salah

1. Sistem memberikan peringatan bahwa data yang diinputkan salah.

2. Kembali ke Alternatif Flow A-2 langkah ke-5



#### 7. PreCondition

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

#### 8. PostConditions

1. Data studio musik di database telah terupdate

### 4.1.5

#### Use case Spesification : Pengelolaan Data Booking

##### 1. Brief Description

Use Case ini memungkinkan operator untuk melakukan pengelolaan data booking, termasuk didalamnya mencari data booking, menampilkan data booking, menghapus data booking, menampilkan data pembayaran, dan membatalkan data booking.

##### 2. Primary Actor

1. Operator

##### 3. Supporting Actor

none

##### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data pembayaran.
2. Sistem memberikan pilihan untuk melakukan pembatalan data booking, pengeditan data pembayaran, serta menampilkan data booking.
3. Aktor memilih untuk melakukan pembatalan booking
  - A-1 Aktor memilih untuk menampilkan data booking
  - A-2 Aktor memilih untuk melakukan pencarian data booking
4. Sistem menampilkan form untuk memasukkan data booking yang akan dibatalkan.
5. Aktor memasukkan data booking yang akan dibatalkan.
6. Sistem melakukan pengecekan terhadap data booking yang diinputkan aktor

- E-1 Data yang diinputkan aktor tidak ditemukan
7. Sistem menampilkan data booking yang ditemukan berdasarkan inputan aktor
  8. Aktor meminta sistem untuk membatalkan data booking yang ditampilkan sistem
  9. Sistem melakukan pembatalan terhadap data booking yang diminta aktor.
  10. Use Case selesai

#### 5. Alternative Flow

- A-1 Aktor memilih untuk menampilkan data booking
1. Sistem menampilkan seluruh data booking yang tersimpan
  2. Berlanjut ke Basic Flow langkah ke-10
- A-2 Aktor memilih untuk melakukan pencarian data booking
1. Sistem menampilkan form untuk memasukkan data booking yang akan dicari
  2. Aktor memasukkan data booking yang ingin dicari
  3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor
- E-2 Data yang diinputkan aktor tidak ditemukan
4. Sistem menampilkan data booking yang dicari aktor
  5. Berlanjut ke Basic Flow langkah ke-10

#### 6. Error Flow

- E-1 Data yang diinputkan aktor tidak ditemukan
1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan
  2. Kembali ke Basic Flow langkah ke-5
- E-2 Data yang diinputkan aktor tidak ditemukan
1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan
  2. Kembali ke Alternative Flow A-2 langkah ke-2

7. **PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

8. **PostConditions**

1. Data booking di database telah terupdate

4.1.6 **Use case Spesification : Pengelolaan Data**  
Pertanyaan

**1. Brief Description**

Use Case ini memungkinkan operator untuk melakukan pengelolaan data pertanyaan, termasuk di dalamnya menjawab pertanyaan, menampilkan data pertanyaan, dan mencari data pertanyaan.

**2. Primary Actor**

1. Operator

**3. Supporting Actor**

None

**4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data pertanyaan.
2. Sistem memberikan pilihan untuk menjawab pertanyaan, pencarian data pertanyaan, atau menampilkan data pertanyaan.
3. Aktor memilih untuk melakukan menjawab pertanyaan
  - A-1 Aktor memilih untuk menampilkan data pertanyaan
  - A-2 Aktor memilih untuk melakukan pencarian data pertanyaan
4. Sistem menampilkan form untuk memasukkan data pertanyaan yang akan dijawab.
5. Aktor memasukkan data pertanyaan yang ingin dijawab
6. Sistem melakukan pengecekan terhadap data yang diinputkan aktor
  - E-1 Data yang diinputkan aktor tidak ditemukan

7. Sistem menampilkan data pertanyaan yang ditemukan berdasarkan inputan aktor
8. Aktor mengubah data pertanyaan yang ditampilkan sistem
9. Sistem melakukan pengecekan terhadap jawaban yang diinputkan aktor

E-2 Data yang diinputkan salah

10. Aktor meminta sistem untuk menyimpan jawaban pertanyaan yang telah dimasukkan
11. Sistem menyimpan jawaban pertanyaan berdasarkan inputan aktor
12. Use Case selesai

## 5. Alternative Flow

A-1 Aktor memilih untuk menampilkan data pertanyaan

1. Sistem menampilkan seluruh data pertanyaan yang tersimpan
2. Berlanjut ke Basic Flow langkah ke-12

A-2 Aktor memilih untuk melakukan pencarian data pertanyaan

1. Sistem menampilkan form untuk memasukkan data pertanyaan yang akan dicari
2. Aktor memasukkan data pertanyaan yang ingin dicari
3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor

E-3 Data yang diinputkan aktor tidak ditemukan

4. Sistem menampilkan data pertanyaan yang dicari aktor
5. Berlanjut ke Basic Flow langkah ke-12

## 6. Error Flow

E-1 Data yang diinputkan aktor tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan
2. Kembali ke Basic Flow langkah ke-5

Program Studi Teknik Informatika	SKPL – MSBS	32/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-2 Data yang diinputkan salah

1. Sistem memberikan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow langkah ke-8

E-3 Data yang diinputkan aktor tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang diinputkan tidak ditemukan
2. Kembali ke Alternative Flow A-4 langkah ke-2

#### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

1. Data pertanyaan di database telah terupdate

#### **4.1.7 Use case Spesification : Generate Laporan Pemesanan Harian**

##### **1. Brief Description**

Use Case ini digunakan oleh operator untuk melihat laporan pemesanan harian yang tersimpan dalam sistem.

##### **2. Primary Actor**

1. Operator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk generate data pemesanan
2. Sistem menampilkan form untuk memasukkan tanggal pemesanan yang ingin digenerate.
3. Aktor memasukkan tanggal pemesanan yang ingin digenerate.
4. Sistem mengecek inputan tanggal pemesanan yang diinputkan aktor.

E-1 Tanggal yang diinputkan aktor tidak valid

Program Studi Teknik Informatika	SKPL – MSBS	33/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

5. Sistem melakukan generate laporan pemesanan harian

6. Use case selesai

#### **5. Alternative Flow**

none

#### **6. Error Flow**

E-1 Tanggal yang diinputkan tidak valid

1. Sistem memberikan peringatan bahwa tanggal yang diinputkan tidak valid
2. Kembali ke Basic Flow langkah ke-3

#### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

1. Laporan pemesanan harian ditampilkan

### **4.1.8 Use case Spesification : Registrasi member**

#### **1. Brief Description**

Use Case ini memungkinkan nonmember untuk melakukan pendaftaran data diri sebagai member.

#### **2. Primary Actor**

1. Nonmember

#### **3. Supporting Actor**

none

#### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan registrasi sebagai member.
2. Sistem menampilkan form untuk memasukkan data member.
3. Aktor memasukkan data diri untuk menjadi member.

4. Sistem melakukan pengecekan terhadap data member yang diinputkan aktor

E-1 Data yang diinputkan salah

E-2 Data yang diinputkan sudah ada

5. Aktor meminta sistem untuk menyimpan data yang telah diinputkan

6. Sistem menyimpan data yang telah diinputkan aktor.

7. Use Case selesai

#### 5. Alternative Flow

none

#### 6. Error Flow

E-1 Data yang diinputkan salah

2. Sistem memberikan peringatan bahwa data yang diinputkan salah

3. Kembali ke Basic Flow langkah ke-3

E-2 Data yang diinputkan sudah ada

1. Sistem memberikan peringatan bahwa data yang diinputkan sudah ada

2. Kembali ke Basic Flow langkah ke-3

#### 7. PreConditions

none

#### 8. PostConditions

Data member di database telah tersimpan

### 4.1.9 Use case Spesification : ubah profil member

#### 1. Brief Description

Use Case ini digunakan oleh member untuk mengubah profil.

#### 2. Primary Actor

1. Member

#### 3. Supporting Actor

none

#### 4. Basic Flow

Program Studi Teknik Informatika	SKPL – MSBS	35/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use Case ini dimulai ketika aktor memilih untuk mengubah profil member
2. Sistem menampilkan antarmuka untuk mengubah profil member
3. Aktor memasukkan profil member yang baru
4. Sistem memeriksa inputan aktor
  - E-1 Inputan tidak valid
5. Aktor meminta sistem untuk menyimpan profil member yang telah diubah
6. Sistem menyimpan profil member berdasarkan ubahan aktor
7. Use Case ini selesai

#### **5. Alternative Flow**

none

#### **6. Error Flow**

E-1 Inputan tidak valid

1. Sistem menampilkan peringatan bahwa inputan tidak valid
2. Kembali ke Basic Flow langkah ke-3

#### **7. PreConditions**

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

Profil member dalam sistem berhasil diubah.

### **4.1.10 Use case Spesification : pengiriman pertanyaan**

#### **1. Brief Description**

Use Case ini digunakan oleh member maupun non-member untuk mengirimkan pertanyaan, kritik, maupun saran kepada manajemen studio musik.

#### **2. Primary Actor**

1. Member
2. Nonmember

Program Studi Teknik Informatika	SKPL – MSBS	36/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



### 3. Supporting Actor

None.

### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengirimkan pertanyaan
2. Sistem menampilkan antarmuka untuk mengirimkan pertanyaan
3. Aktor memasukkan data pertanyaan yang hendak dikirim
4. Sistem memeriksa inputan aktor  
E-1 Inputan tidak valid
5. Aktor meminta sistem untuk mengirimkan data pertanyaan
6. Sistem menyimpan data pertanyaan yang diinputkan aktor
7. Use Case ini selesai

### 5. Alternative Flow

none

### 6. Error Flow

E-1 Inputan tidak valid

1. Sistem menampilkan peringatan bahwa inputan tidak valid
2. Kembali ke Basic Flow langkah ke-3

### 7. PreConditions

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

### 8. PostConditions

Pertanyaan telah disimpan dalam sistem

#### 4.1.11

#### Use case Spesification : Booking studio

##### 1. Brief Description

Use Case ini digunakan oleh member untuk melakukan booking atau pemesanan studio musik.

##### 2. Primary Actor

1. member

##### 3. Supporting Actor

Program Studi Teknik Informatika	SKPL – MSBS	37/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

None

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan booking studio musik secara online.
2. Sistem menampilkan jadwal-jadwal yang tersedia di studio musik.
3. Aktor memilih jadwal-jadwal yang hendak dipesan.
4. Sistem menampilkan data booking sementara dan memberikan pilihan kepada aktor untuk memilih melanjutkan proses booking atau memilih untuk mengubah data booking studio musik.
5. Aktor memilih untuk melakukan pemesanan
  - A-1 Aktor memilih untuk melanjutkan proses booking studio musik
  - A-2 Aktor memilih untuk mengubah data booking
6. Sistem menampilkan formulir booking
7. Aktor mengisi formulir booking yang ditampilkan sistem
8. Sistem melakukan pengecekan terhadap data booking yang diinputkan aktor
  - E-1 Data yang diinputkan salah
9. Aktor meminta sistem untuk melakukan penyimpanan terhadap data booking diinputkan
10. Sistem menyimpan data yang telah diinputkan aktor.
11. Use Case selesai

#### 5. Alternative Flow

- A-1 Aktor memilih untuk melanjutkan proses booking
  1. Berlanjut ke Basic Flow langkah ke-2
- A-2 Aktor memilih untuk mengubah data booking
  1. Sistem menampilkan data booking sementara
  2. Aktor mengubah data booking sementara yang ditampilkan sistem

3. Sistem melakukan pengecekan terhadap data yang diinputkan aktor

E-2 Data yang diinputkan salah

4. Aktor meminta sistem untuk menyimpan data booking sementara yang telah diubah

5. Sistem melakukan penyimpanan terhadap data booking sementara yang diubah aktor

6. Berlanjut ke Basic Flow langkah ke-4

#### **6. Error Flow**

E-1 Data yang diinputkan salah

1. Sistem memberikan peringatan bahwa data yang diinputkan salah

2. Kembali ke Basic Flow langkah ke-7

E-2 Data yang diinputkan salah

1. Sistem memberikan peringatan bahwa data yang diinputkan salah

2. Kembali ke Alternative Flow A-2 langkah ke-2

#### **7. PreConditions**

Booking dilakukan oleh user yang telah menjadi member.

#### **8. PostConditions**

Data booking telah tersimpan di database

### **4.1.12**

**Use case Spesification : Lihat Jadwal**

#### **1. Brief Description**

Use Case ini digunakan oleh member dan non-member untuk melihat jadwal studio music yang sudah dipesan maupun yang belum dipesan.

#### **2. Primary Actor**

1. Member

2. Non-Member

#### **3. Supporting Actor**

None

Program Studi Teknik Informatika	SKPL – MSBS	39/ 42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melihat jadwal studio musik secara online.
2. Sistem menampilkan kalender yang tersedia pada form jadwal.
3. Aktor memasukkan tanggal yang ingin dilihat.

E-1 Data yang diinputkan salah

4. Sistem menampilkan jadwal-jadwal yang tersedia di studio musik.
5. Use Case selesai

#### 5. Alternative Flow

1. None

#### 6. Error Flow

E-1 Data yang diinputkan salah

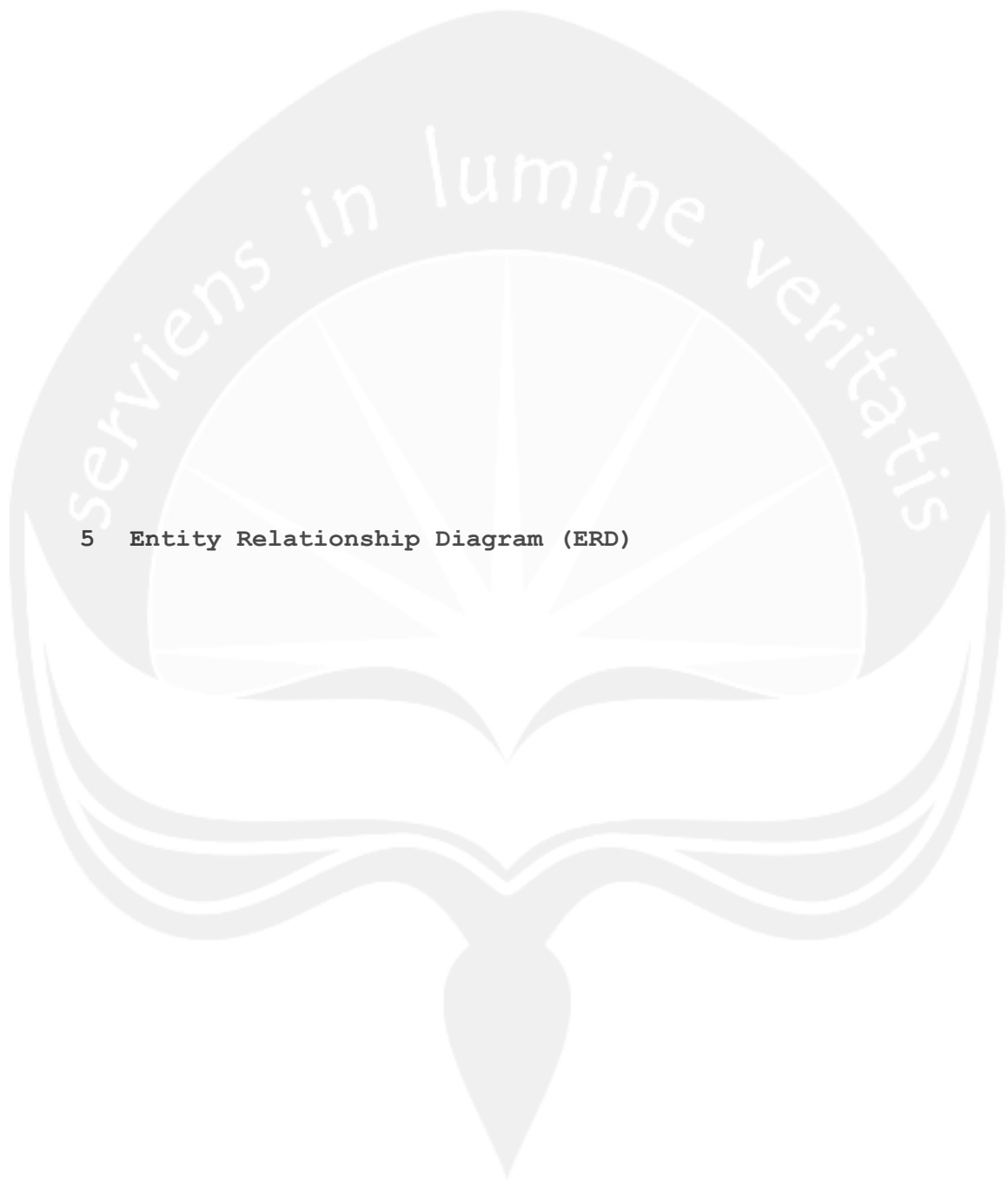
1. Sistem memberikan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow langkah ke-7
3. Sistem memberikan peringatan bahwa data yang diinputkan salah
4. Kembali ke Basic Flow langkah ke-7

#### 7. PreConditions

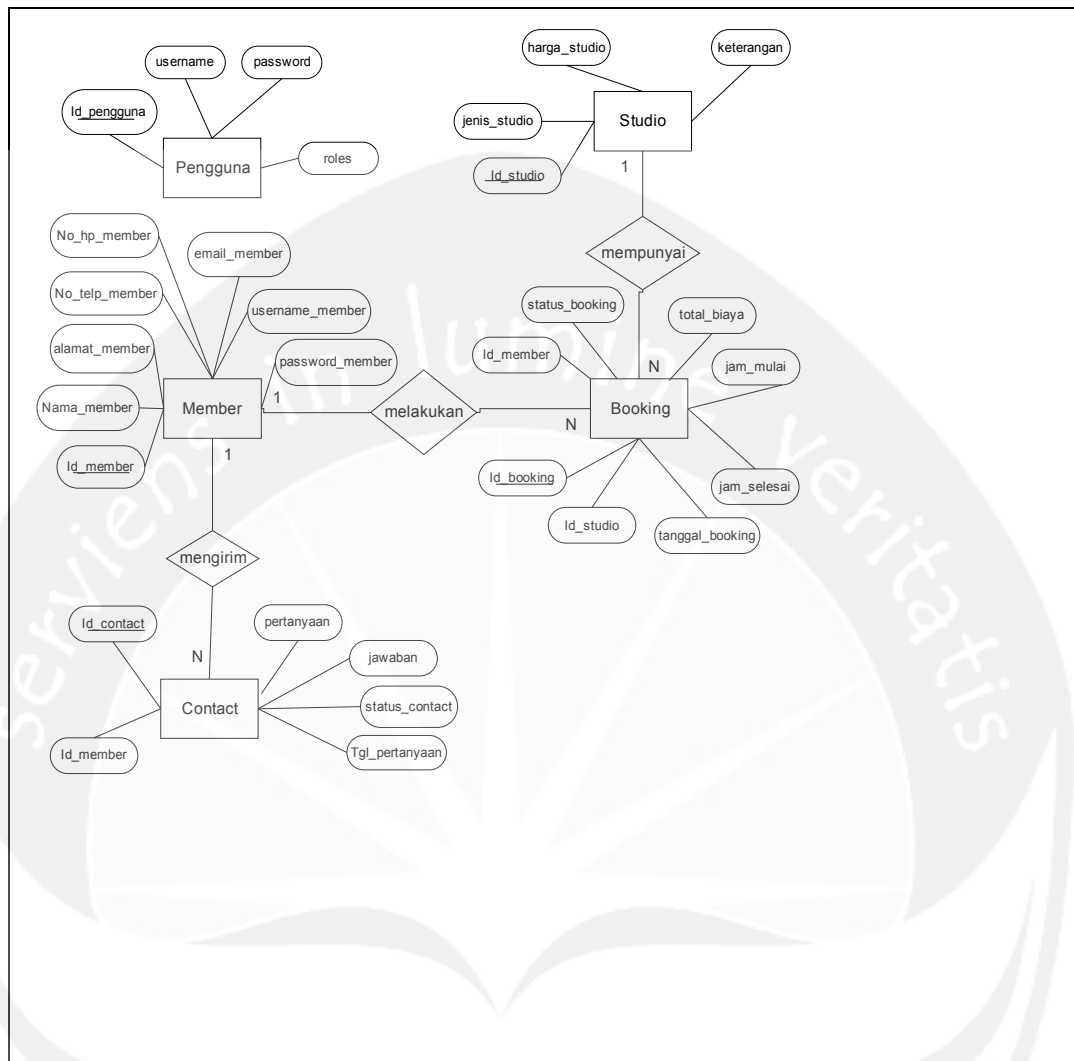
1. None

#### 8. PostConditions

Data jadwal yang tersimpan pada database telah ditampilkan



## 5 Entity Relationship Diagram (ERD)



**Gambar 3. Entity Relationship Diagram (ERD)**

# DPPL

## DESKRIPSI PERANCANGAN PERANGKAT LUNAK

### MSBS

(Music Studio Booking System)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Bagus Perwira Laksmana / 5177

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen	Halaman
		<b>DPPL-MSBS</b>	1/110

Program Studi Teknik Informatika	DPPL - MSBS	1/54
----------------------------------	-------------	------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

	Fakultas Teknologi Industri	Revisi		
--	-----------------------------	--------	--	--

### DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								



### Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



## Daftar Isi

1 Pendahuluan.....	8
1.1 Tujuan.....	8
1.2 Ruang Lingkup.....	8
1.3 Definisi dan Akronim.....	8
1.4 Referensi.....	9
2 Perancangan Sistem.....	10
2.1 Perancangan Arsitektur.....	10
2.2 Perancangan Rinci.....	11
2.2.1 Sequence Diagram.....	11
2.2.2 Diagram Kelas.....	25
2.2.2.1 Deskripsi Kelas.....	26
3 Perancangan Data.....	38
3.1 Dekomposisi Data.....	38
3.1.1 Deskripsi Entitas Data Pengguna.....	38
3.1.2 Deskripsi Entitas Data Studio.....	38
3.1.3 Deskripsi Entitas Data Member.....	38
3.1.4 Deskripsi Entitas Data Booking.....	38
3.1.4.1 id_booking.....	39
3.1.5 Deskripsi Entitas Data Contact.....	39
3.2 PDM.....	40
4 Perancangan Antarmuka.....	41

## Daftar Gambar

Gambar 2.1 Perancangan Arsitektur.....	13
Gambar 2.2 Sequence Diagram : Login.....	14
Gambar 2.3 Sequence Diagram : Pengelolaan Data Pengguna - Tambah Pengguna.....	14
Gambar 2.4 Sequence Diagram : Pengelolaan Data Pengguna - Edit Pengguna.....	15
Gambar 2.5 Sequence Diagram : Pengelolaan Data Pengguna - Hapus Pengguna.....	15
Gambar 2.6 Sequence Diagram : Pengelolaan Data Pengguna - Tampil Pengguna.....	16
Gambar 2.7 Sequence Diagram : Pengelolaan Data Pengguna - Cari Pengguna.....	16
Gambar 2.8 Sequence Diagram : Pengelolaan Data Member - Hapus Member.....	17
Gambar 2.9 Sequence Diagram : Pengelolaan Data Member - Tampil Member.....	17
Gambar 2.10 Sequence Diagram : Pengelolaan Data Member - Cari Member.....	18
Gambar 2.11 Sequence Diagram : Pengelolaan Data Studio - Tambah Studio.....	18
Gambar 2.12 Sequence Diagram : Pengelolaan Data Studio - Edit Studio.....	19
Gambar 2.13 Sequence Diagram : Pengelolaan Data Studio - Hapus Studio.....	19
Gambar 2.14 Sequence Diagram : Pengelolaan Data Studio - Tampil Studio.....	20
Gambar 2.15 Sequence Diagram : Pengelolaan Data Studio - Cari Studio.....	20
Gambar 2.16 Sequence Diagram : Pengelolaan Data Booking - Batal Booking.....	21
Gambar 2.17 Sequence Diagram : Pengelolaan Data Booking g - Tampil Booking.....	21

Gambar 2.18 Sequence Diagram : Pengelolaan Data Booking - Cari Booking.....	22
Gambar 2.19 Sequence Diagram : Pengelolaan Data Pertanyaan - Jawab Pertanyaan.....	22
Gambar 2.20 Sequence Diagram : Pengelolaan Data Pertanyaan - Tampil Pertanyaan.....	23
Gambar 2.21 Sequence Diagram : Pengelolaan Data Pertanyaan - Cari Pertanyaan.....	23
Gambar 2.22 Sequence Diagram : Generate Laporan Harian..	24
Gambar 2.23 Sequence Diagram : Registrasi Member.....	24
Gambar 2.24 Sequence Diagram : Ubah Profil Member.....	25
Gambar 2.25 Sequence Diagram : Kirim Pertanyaan.....	25
Gambar 2.26 Sequence Diagram : Booking Studio.....	26
Gambar 2.57 Class Diagram.....	27
Gambar 3.1 PDM.....	42
Gambar 4.1 Perancangan Antarmuka - Login Member.....	43
Gambar 4.2 Perancangan Antarmuka - Login Pengelolaan.....	43
Gambar 4.3 Perancangan Antarmuka - Pengelolaan Data Pegguna.....	44
Gambar 4.4 Perancangan Antarmuka - Pengelolaan Data Member.....	45
Gambar 4.5 Perancangan Antarmuka - Pengelolaan Data Pertanyaan.....	46
Gambar 4.6 Perancangan Antarmuka - Pengelolaan Data Booking.....	47
Gambar 4.7 Perancangan Antarmuka - Pengelolaan Data Studio.....	48
Gambar 4.8 Perancangan Antarmuka - Tampil Laporan Harian.....	50
Gambar 4.9 Perancangan Antarmuka - Registrasi Member.....	51
Gambar 4.10 Perancangan Antarmuka - Ubah Profil Member...	52
Gambar 4.11 Perancangan Antarmuka - Kirim Pertanyaan.....	53
Gambar 4.12 Perancangan Antarmuka - Booking Studio.....	54

## 1 Pendahuluan

### 1.1 Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

### 1.2 Ruang Lingkup

Perangkat Lunak MSBS dikembangkan dengan tujuan untuk :

1. Menangani Login.
2. Menangani pengelolaan data pengguna sistem.
3. Menangani pengelolaan data member.
4. Menangani pengelolaan data studio musik.
5. Menangani pengelolaan data booking.
6. Menangani pengelolaan data pertanyaan.
7. Fungsi Generate Laporan Pemesanan Harian.
8. Menangani registrasi member.
9. Menangani pengubahan profil member.
10. Menangani pengiriman pertanyaan, kritik, dan saran dari member EHA Music Studio kepada pihak manajemen.
11. Menangani booking studio secara online.
12. Menangani lihat jadwal studio musik secara online.

### 1.3 Definisi dan Akronim

Daftar definisi akronim dan singkatan :

Tabel 1.1 Daftar Definisi Akronim dan Singkatan

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD) merupakan deskripsi dari perancangan produk/perangkat lunak yang akan dikembangkan.

MSBS	Perangkat lunak yang dikembangkan untuk membantu <i>pihak studio musik</i> dalam menangani booking studio berbasis web.
------	---

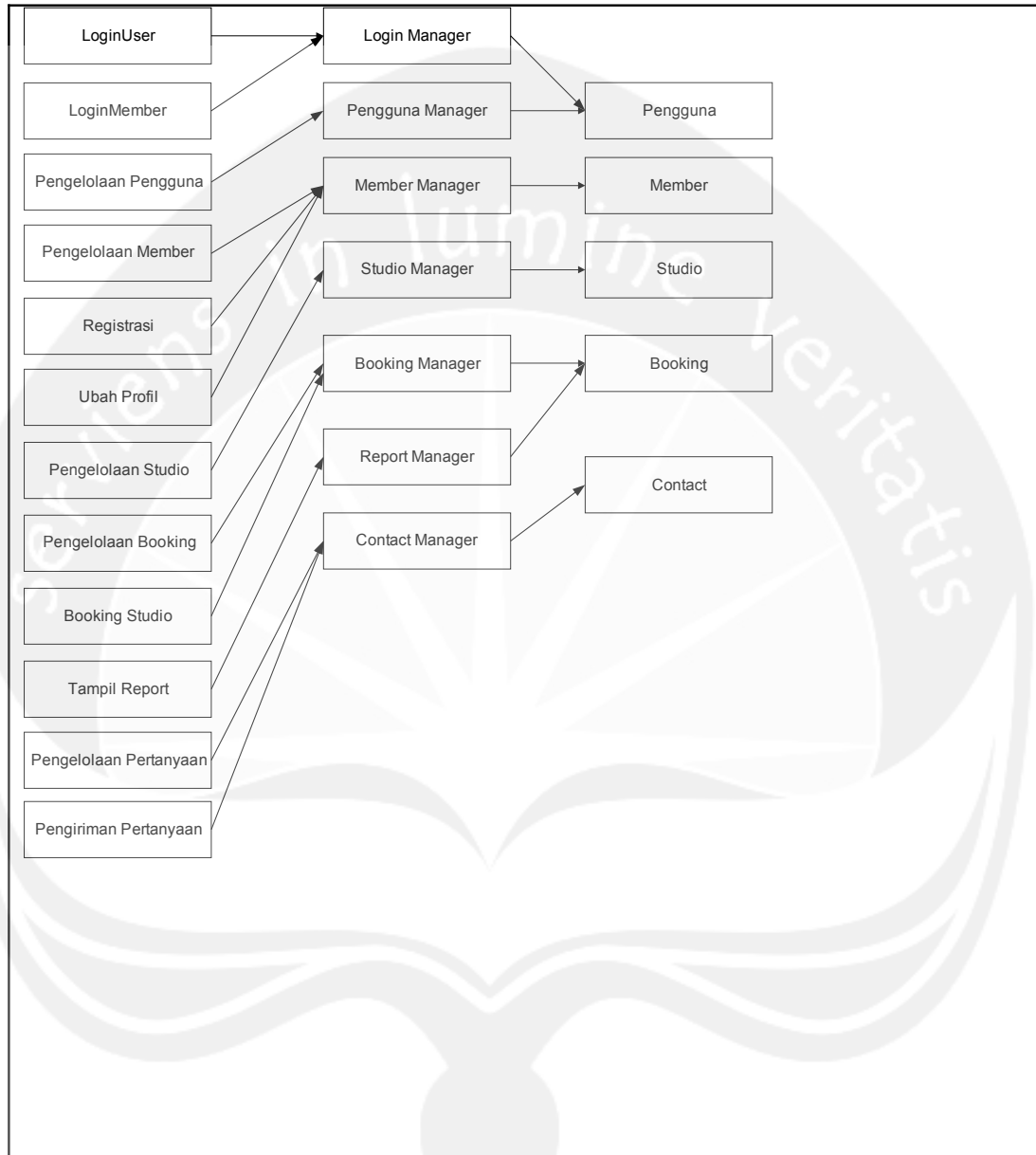
#### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Juli Sapta Putra Hantana, Sepesifikasi Kebutuhan Perangkat Lunak SC3(Smart Client for Cyber Community), Universitas Atmajaya Yogyakarta.
2. Perwira Laksmana, Bagus, Spesifikasi Kebutuhan Perangkat Lunak MSBS (Music Studio Booking System), 2012.

## 2 Perancangan Sistem

### 2.1 Perancangan Arsitektur



Gambar 2.1 Perancangan Arsitektur

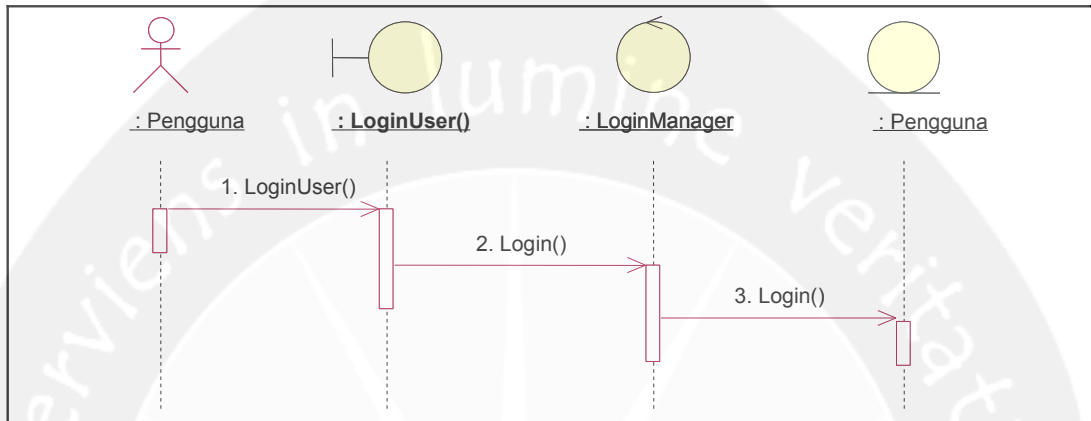


## 2.2 Perancangan Rinci

### 2.2.1 Sequence Diagram

#### 2.2.1.1 Login User

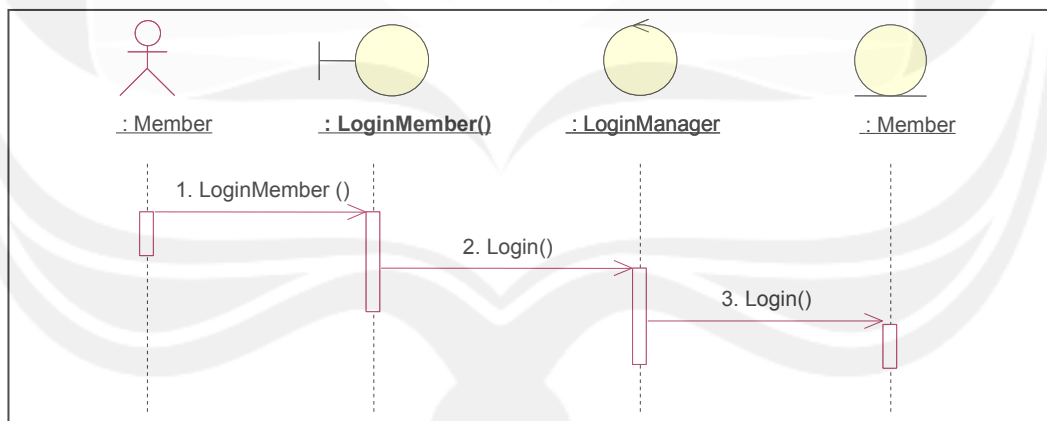
Pengguna yang dimaksud di sini adalah Administrator, Manager, Operator.



Gambar 2.2 Sequence Diagram : LoginUser

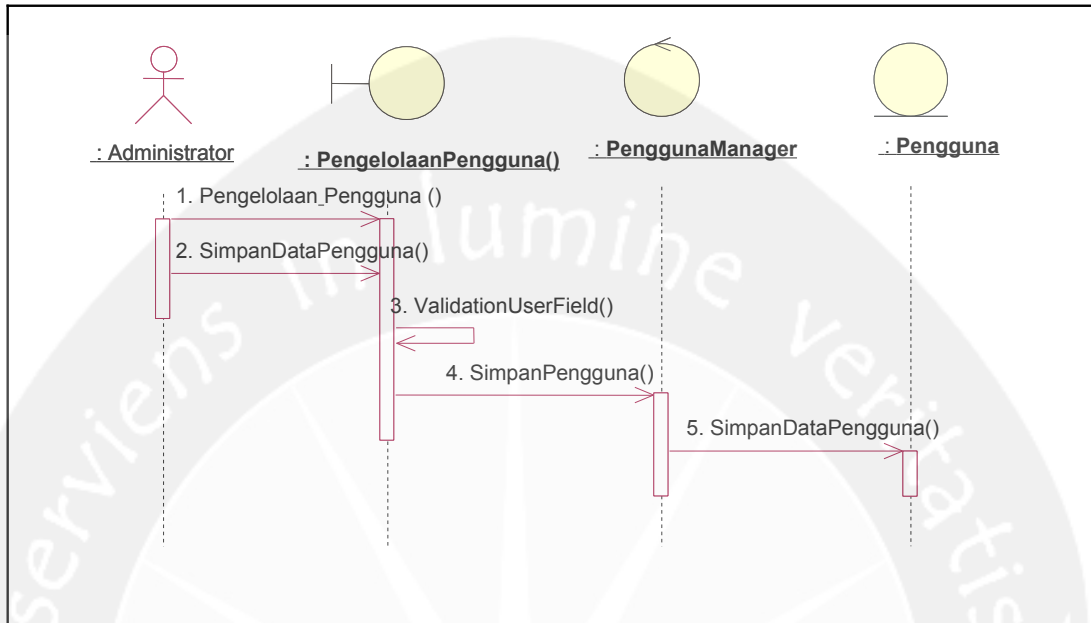
#### 2.2.1.2 Login Member

Login ini merupakan login yang digunakan oleh member.



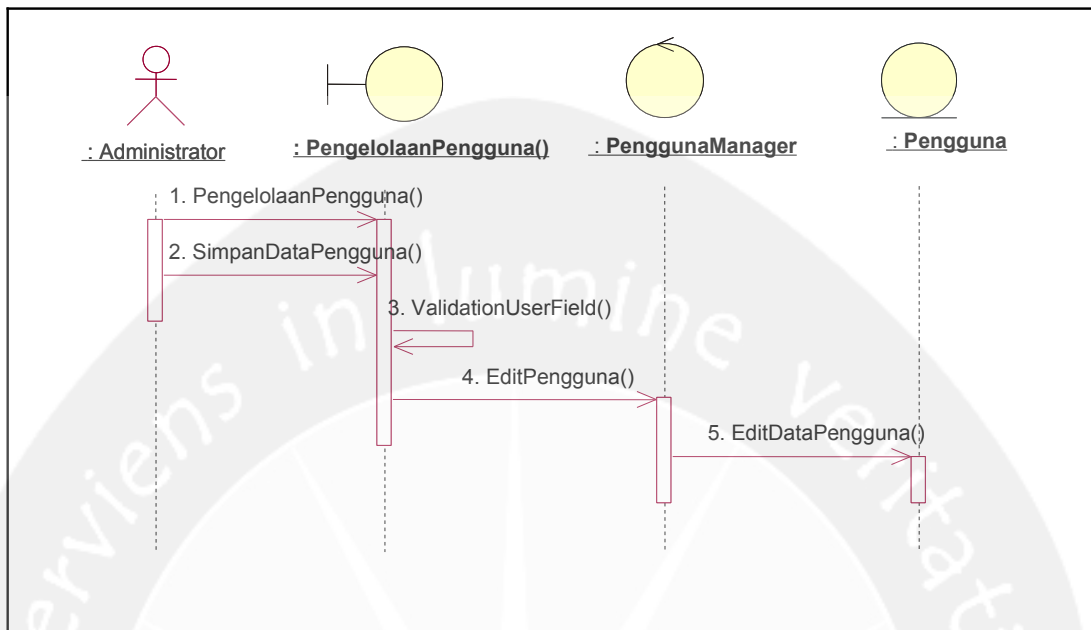
### 2.2.1.3 Pengelolaan Data Pengguna

#### 2.2.1.3.1 Tambah Pengguna



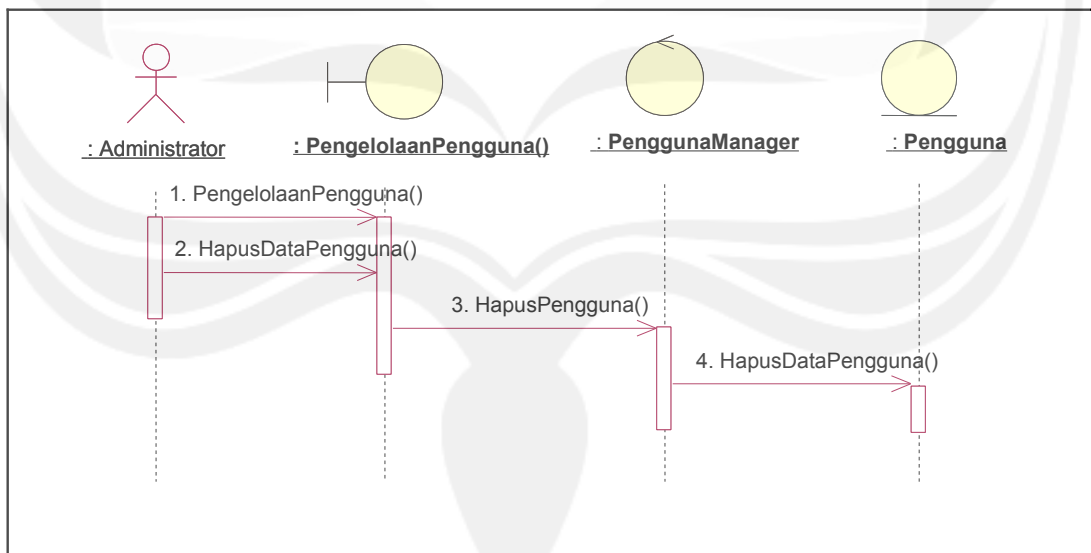
Gambar 2.3 Sequence Diagram : Pengelolaan Data Pengguna – Tambah Pengguna

### 2.2.1.3.2 Edit Pengguna



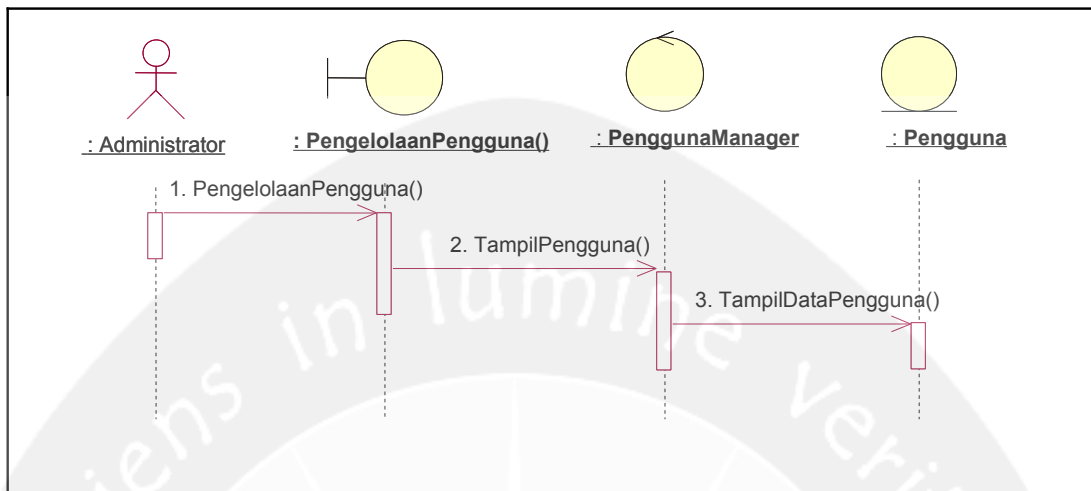
Gambar 2.4 Sequence Diagram : Pengelolaan Data Pengguna - Edit Pengguna

### 2.2.1.3.3 Hapus Pengguna



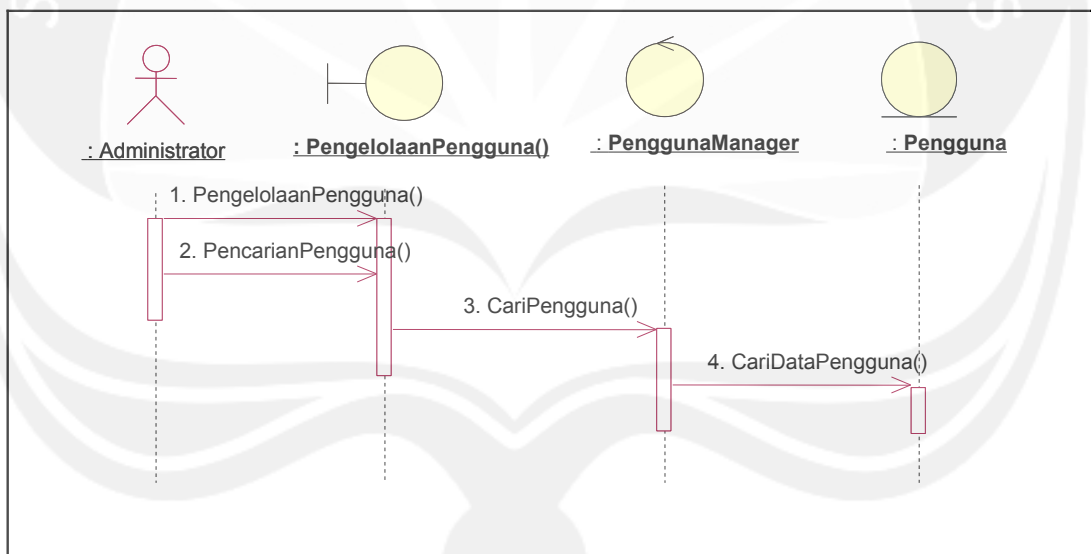
Gambar 2.5 Sequence Diagram : Pengelolaan Data Pengguna - Hapus Pengguna

#### 2.2.1.3.4 Tampil Pengguna



Gambar 2.6 Sequence Diagram : Pengelolaan Data Pengguna - Tampil Pengguna

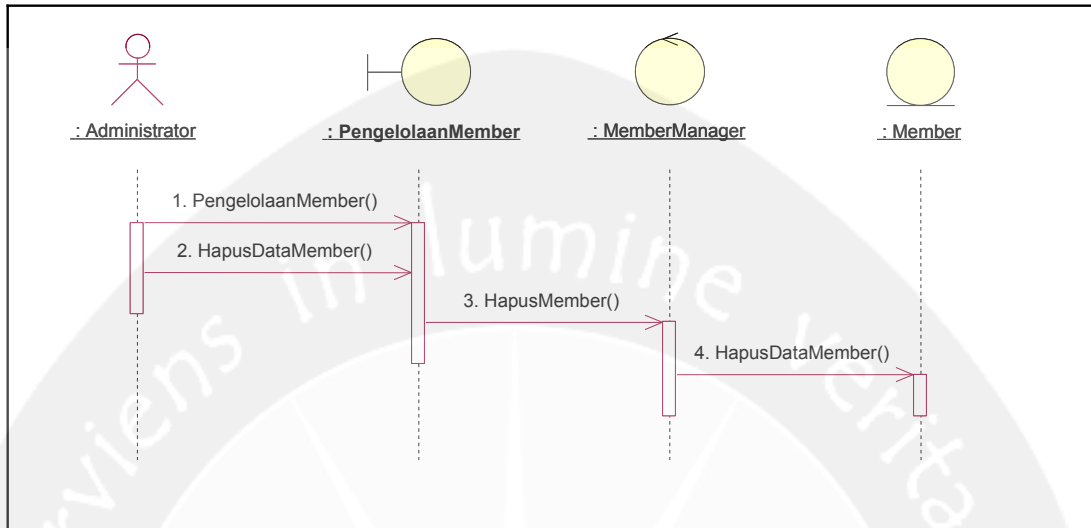
#### 2.2.1.3.5 Cari Pengguna



Gambar 2.7 Sequence Diagram : Pengelolaan Data Pengguna - Cari Pengguna

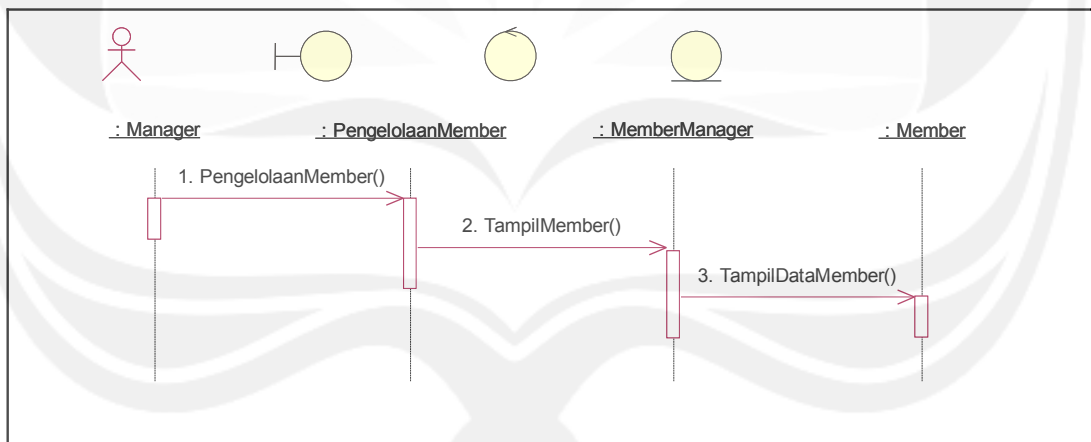
#### 2.2.1.4 Pengelolaan Data Member

##### 2.2.1.4.1 Hapus Member



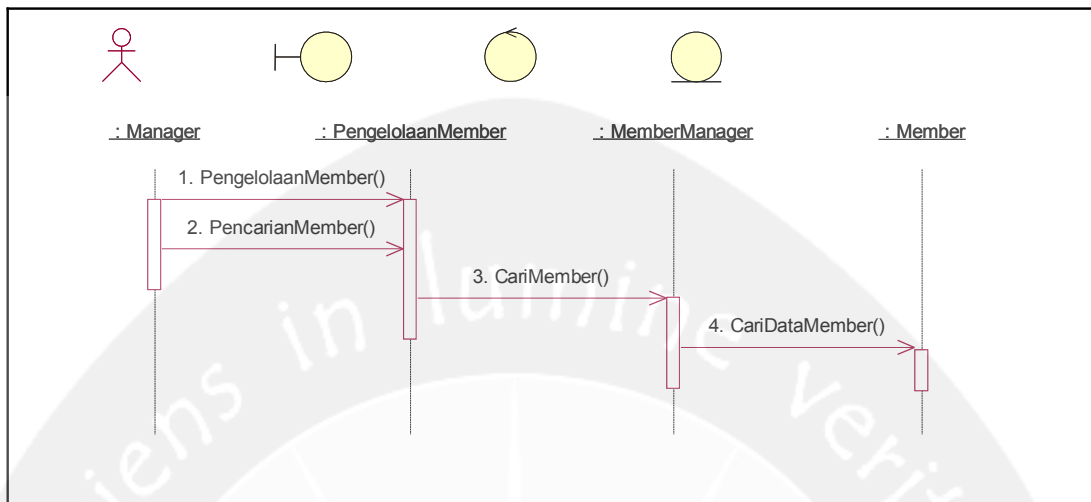
Gambar 2.8 Sequence Diagram : Pengelolaan Data Member  
- Hapus Member

##### 2.2.1.4.2 Tampil Member



Gambar 2.9 Sequence Diagram : Pengelolaan Data Member  
- Tampil Member

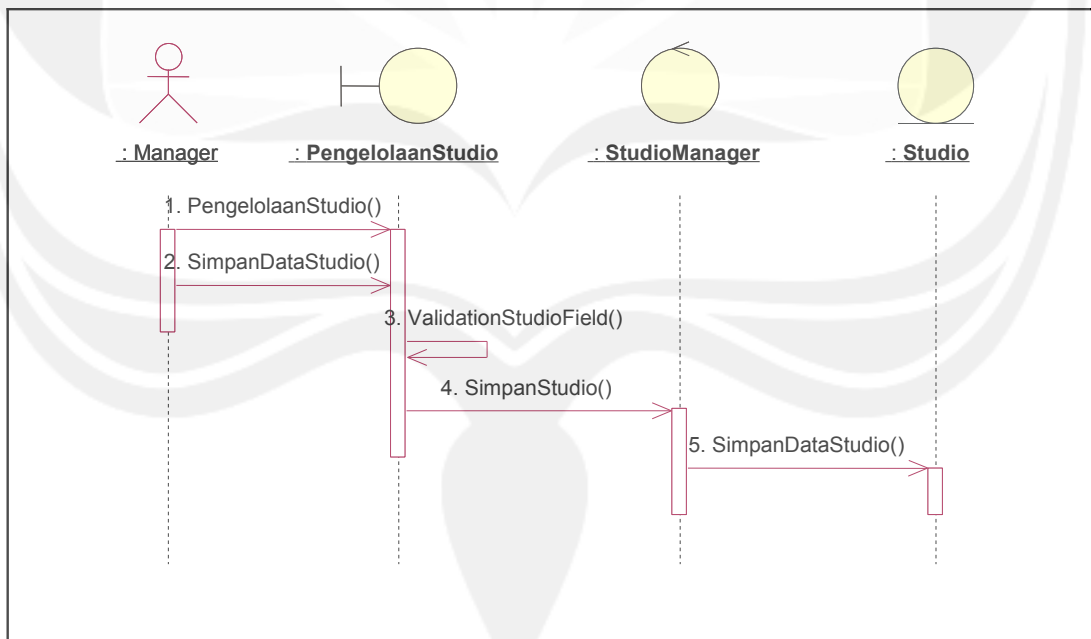
### 2.2.1.4.3 Cari Member



Gambar 2.10 Sequence Diagram : Pengelolaan Data Member - Cari Member

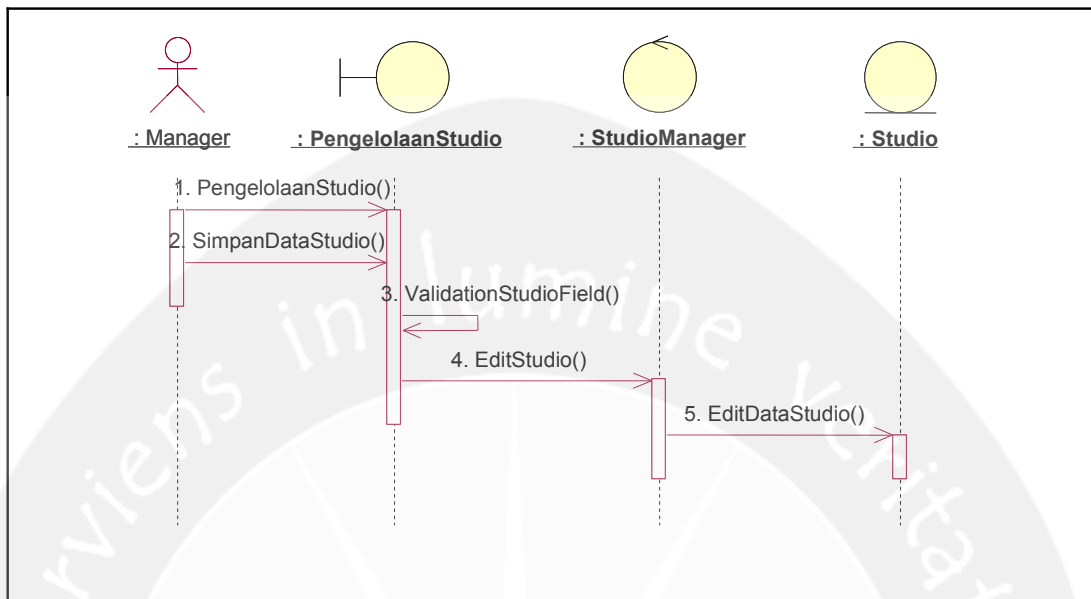
### 2.2.1.5 Pengelolaan Data Studio Musik

#### 2.2.1.5.1 Tambah Studio



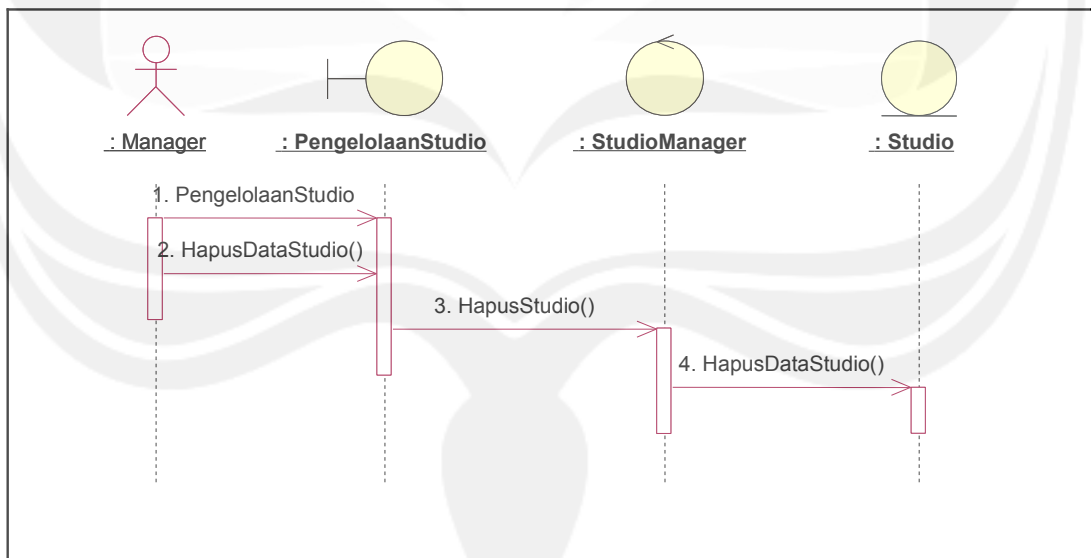
Gambar 2.11 Sequence Diagram : Pengelolaan Data Studio - Tambah Studio

### 2.2.1.5.2 Edit Studio



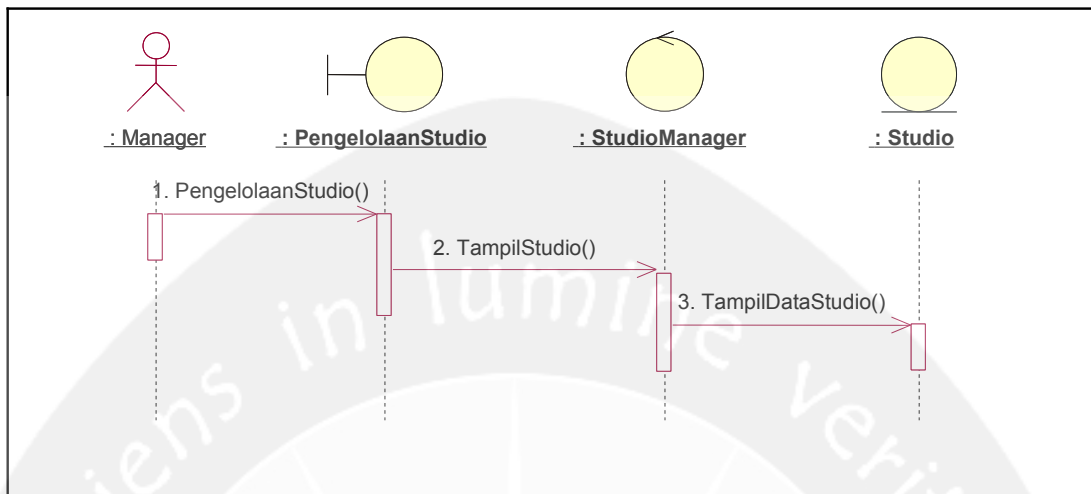
Gambar 2.12 Sequence Diagram : Pengelolaan Data Studio - Edit Studio

### 2.2.1.5.3 Hapus Studio



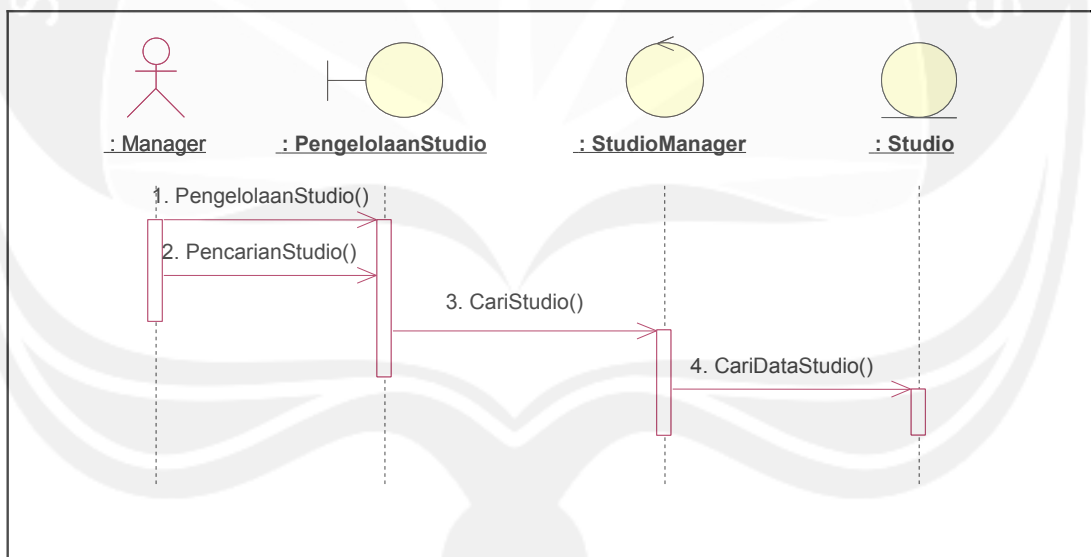
Gambar 2.13 Sequence Diagram : Pengelolaan Data Studio - Hapus Studio

#### 2.2.1.5.4 Tampil Studio



Gambar 2.14 Sequence Diagram : Pengelolaan Data Studio - Tampil Studio

#### 2.2.1.5.5 Cari Studio

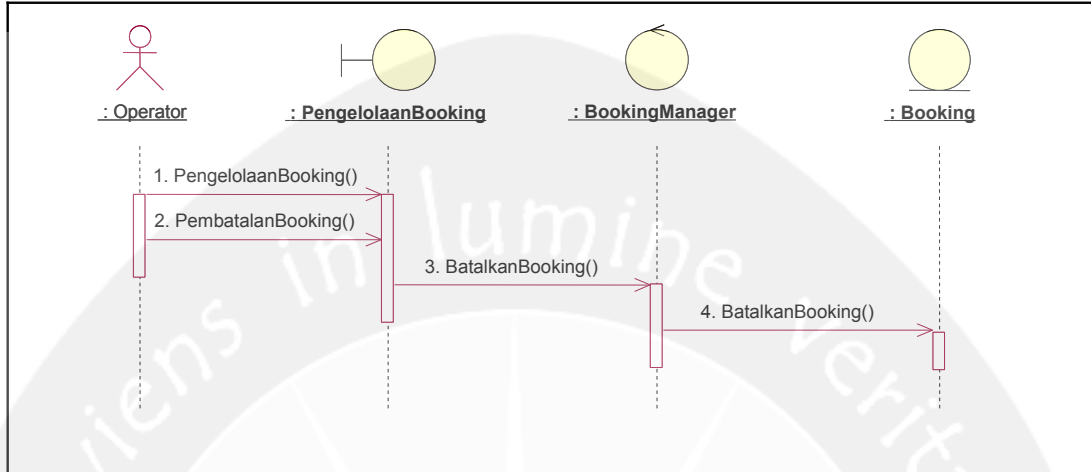


Gambar 2.15 Sequence Diagram : Pengelolaan Data Studio - Cari Studio



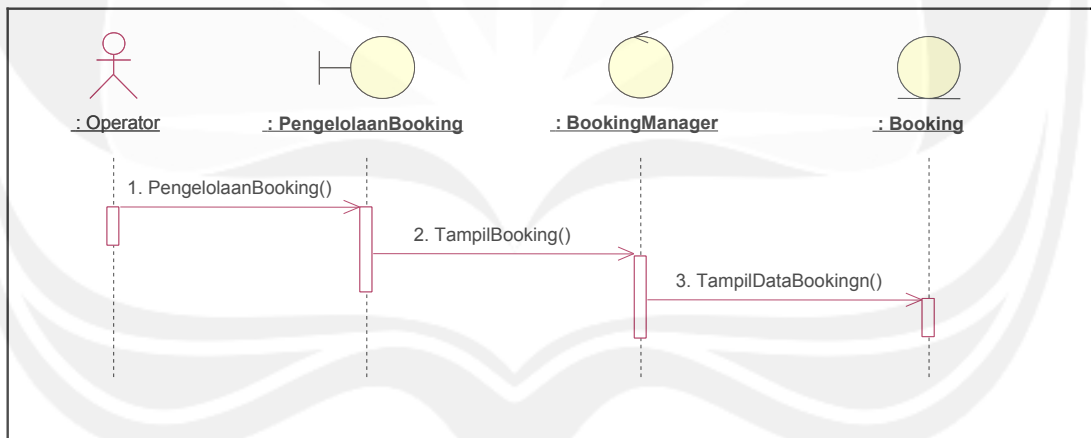
### 2.2.1.6 Pengelolaan Data Booking

#### 2.2.1.6.1 Pembatalan Booking



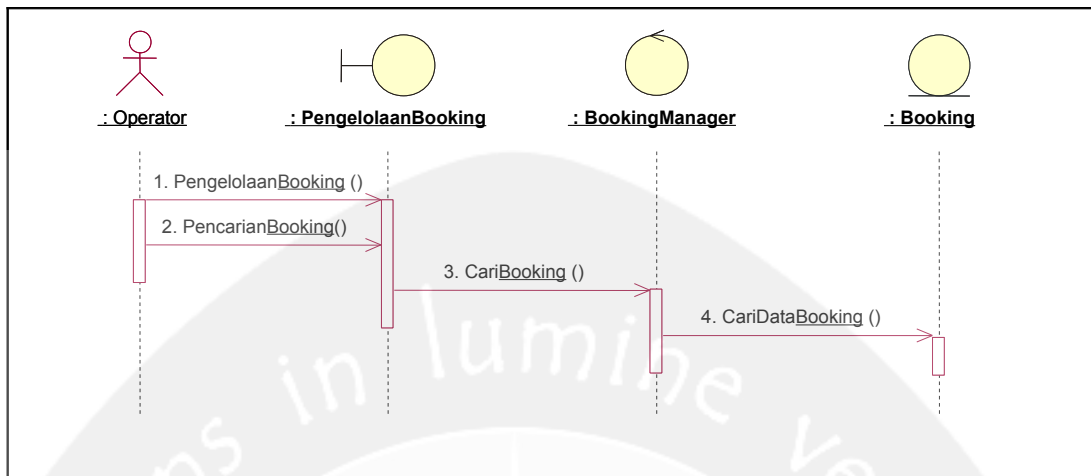
Gambar 2.33 Sequence Diagram : Pengelolaan Data Booking - Pembatalan Booking

#### 2.2.1.6.2 Tampil Booking



Gambar 2.34 Sequence Diagram : Pengelolaan Data Booking - Tampil Booking

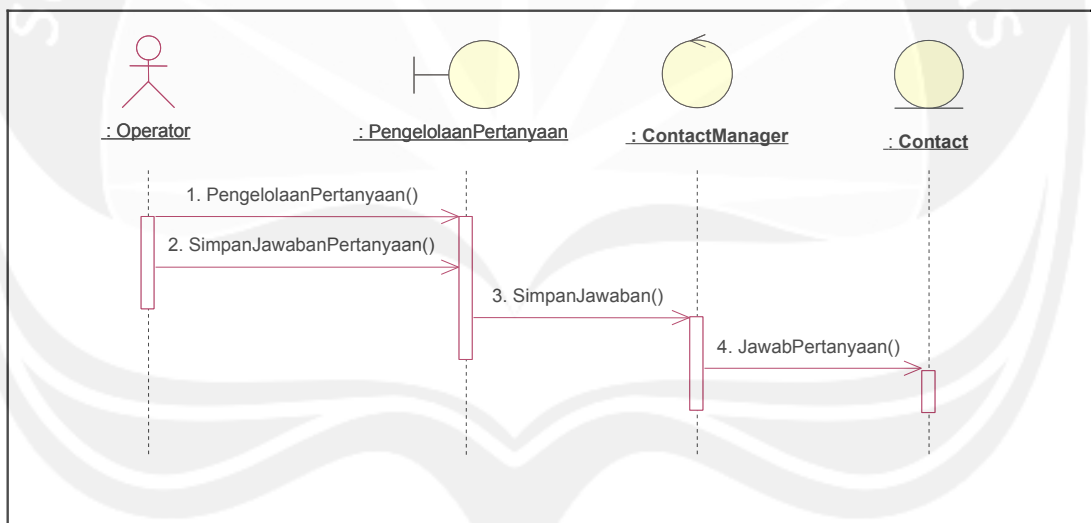
#### 2.2.1.6.3 Cari Booking



Gambar 2.35 Sequence Diagram : Pengelolaan Data Booking - Cari Booking

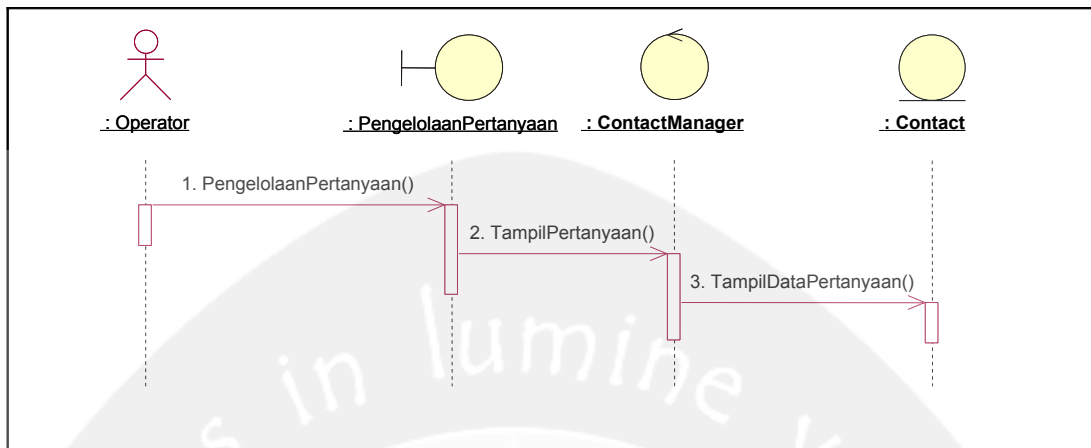
### 2.2.1.7 Pengelolaan Data Pertanyaan

#### 2.2.1.7.1 Jawab Pertanyaan



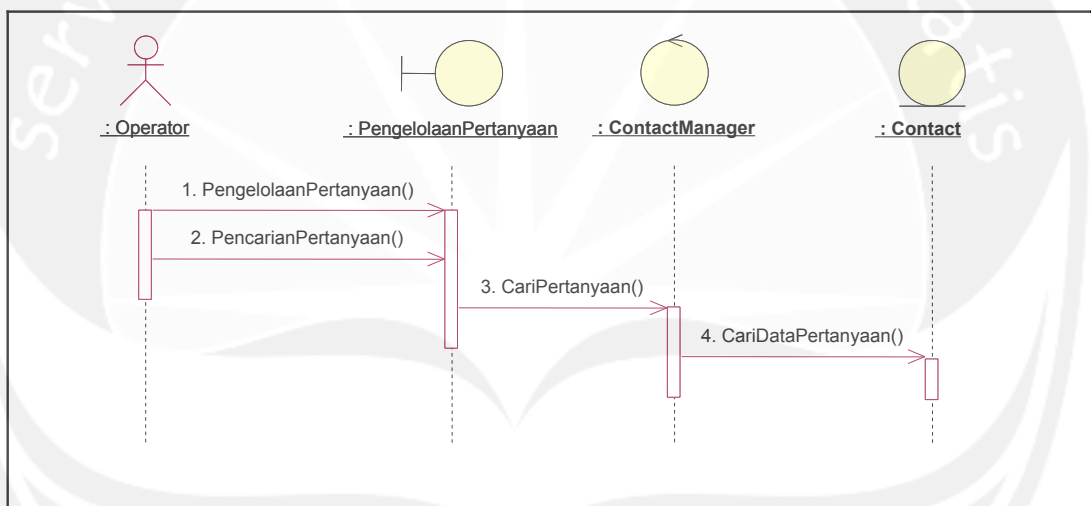
Gambar 2.40 Sequence Diagram : Pengelolaan Data Pertanyaan - Jawab Pertanyaan

#### 2.2.1.7.2 Tampil Pertanyaan



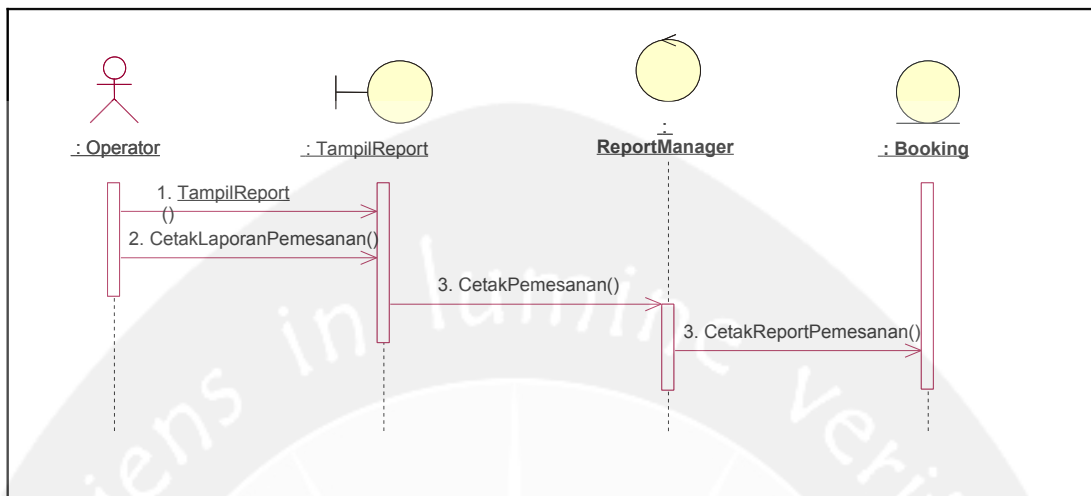
Gambar 2.41 Sequence Diagram : Pengelolaan Data  
Pertanyaan - Tampil Pertanyaan

#### 2.2.1.7.3 Cari Pertanyaan



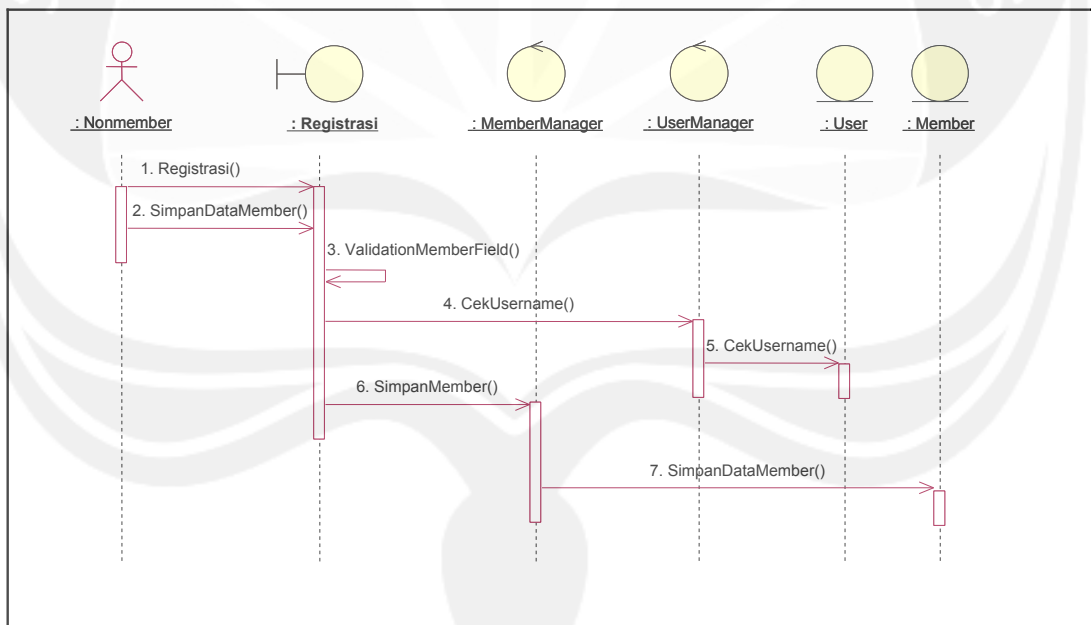
Gambar 2.42 Sequence Diagram : Pengelolaan Data  
Pertanyaan - Cari Pertanyaan

### 2.2.1.8 Generate Laporan Pemesanan Harian



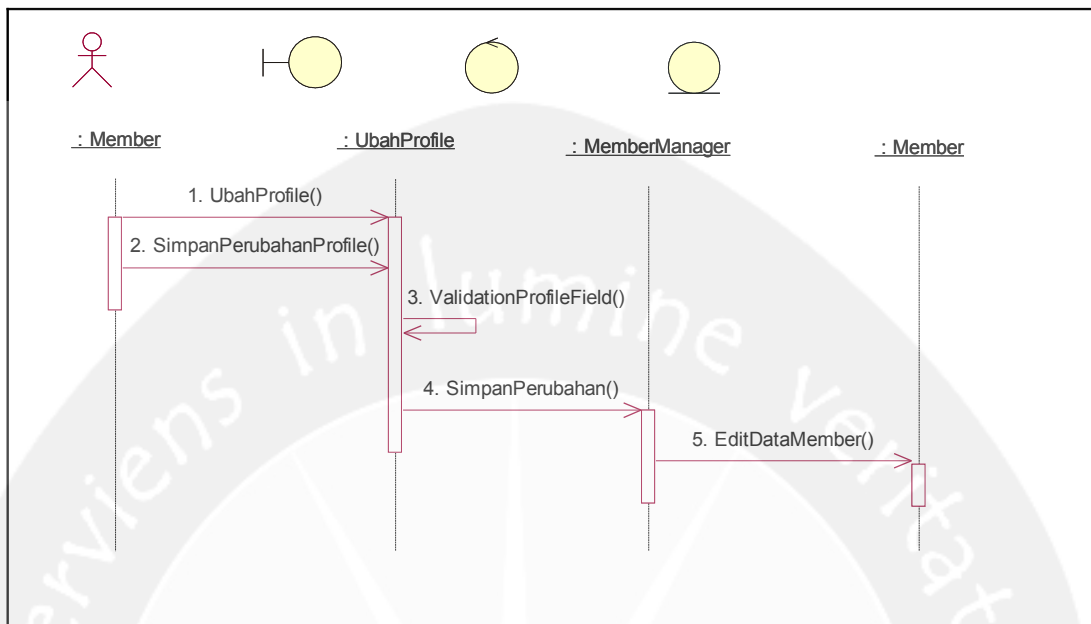
Gambar 2.43 Sequence Diagram : Generate Laporan Pemesanan Harian

### 2.2.1.9 Registrasi Member



Gambar 2.47 Sequence Diagram : Registrasi Member

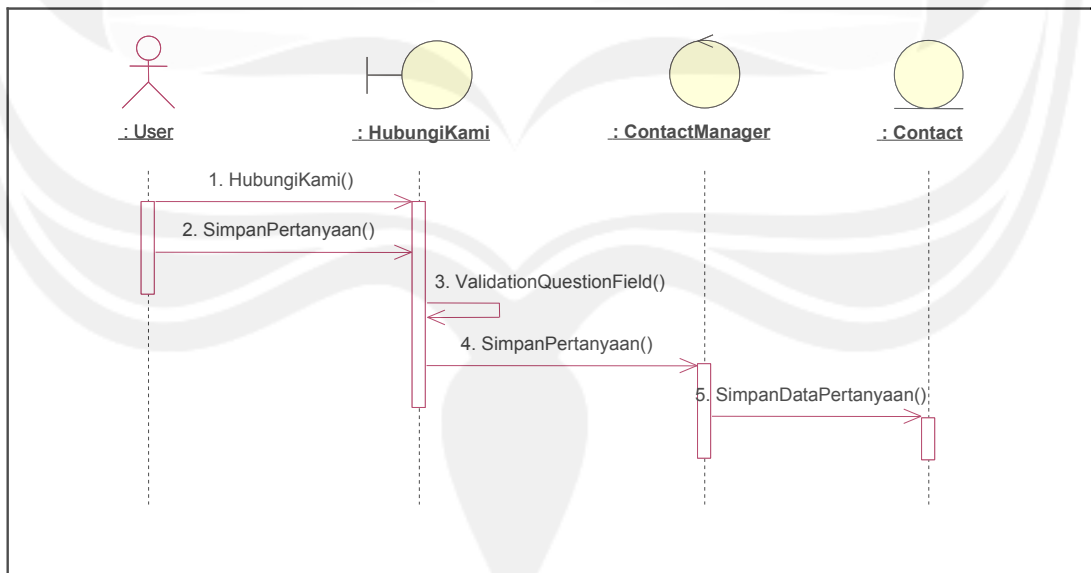
#### 2.2.1.10 Ubah Profil Member



Gambar 2.48 Sequence Diagram : Ubah Profil Member

#### 2.2.1.11 Pengiriman Pertanyaan

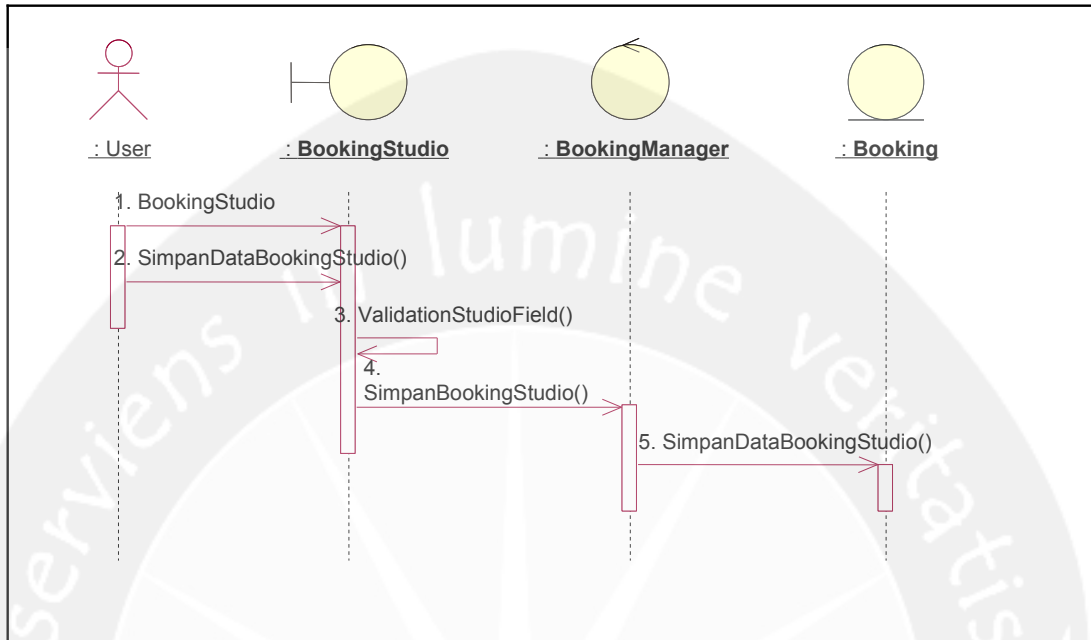
User yang dimaksud di sini adalah Member.



Gambar 2.49 Sequence Diagram : Pengiriman Pertanyaan

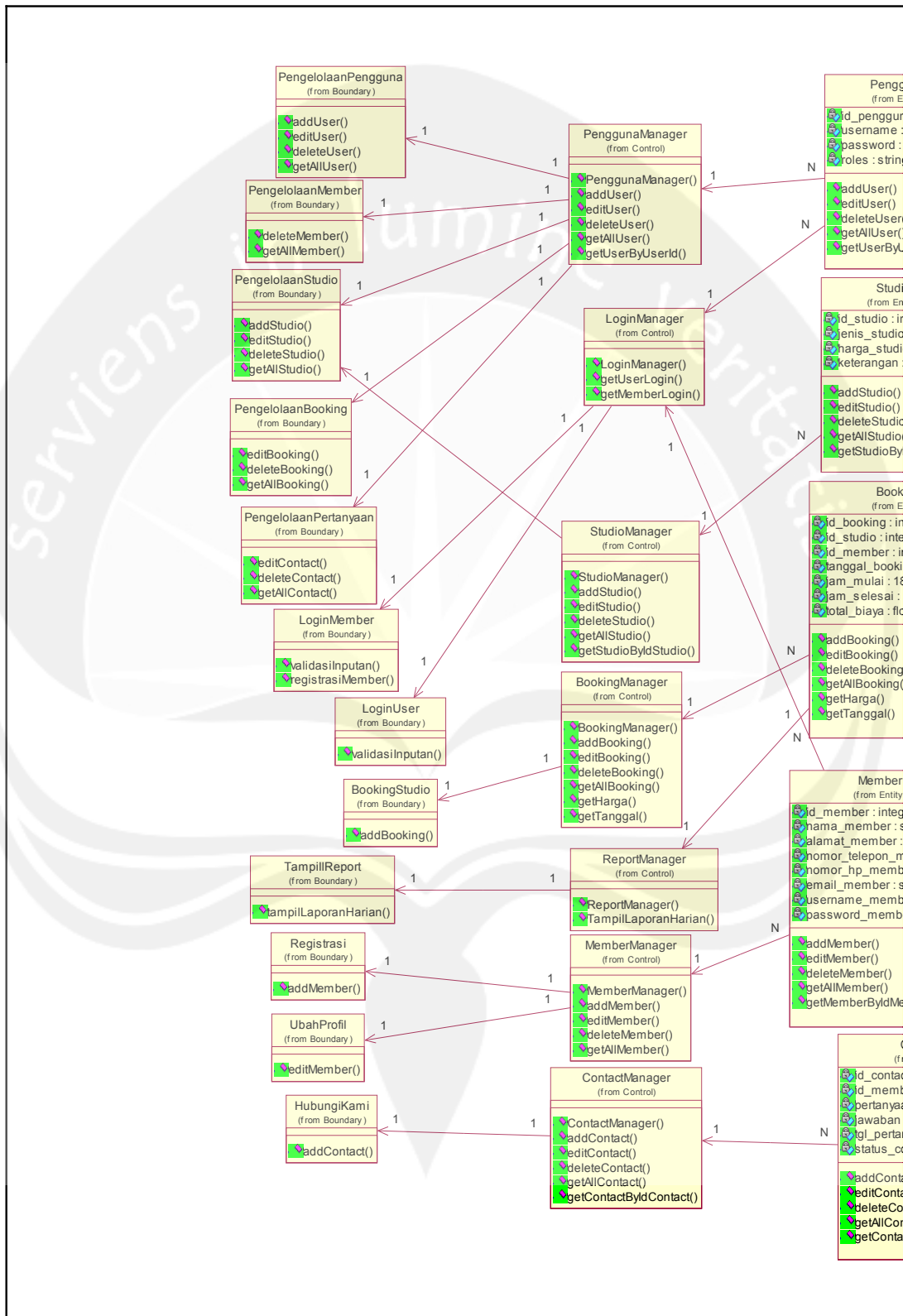
#### 2.2.1.12 Melakukan Booking

User yang dimaksud di sini adalah Member.



Gambar 2.56 Sequence Diagram : Booking Studio

### 2.2.2 Diagram Kelas



Gambar 2.57 Class Diagram

## Deskripsi Kelas

### 2.2.2.1 Specific Design Class LoginUser

LoginUser	<<boundary>>
+ LoginUser() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	

### 2.2.2.2 Specific Design Class LoginMember

LoginMember	<<boundary>>
+ LoginMember() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	

### 2.2.2.3 Specific Design Class PengelolaanPengguna

PengelolaanPengguna	<<boundary>>
+ PengelolaanPengguna() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+ SimpanDataPengguna() Fungsi yang digunakan untuk memproses penyimpanan data Pengguna.	
+ HapusDataPengguna() Fungsi yang digunakan untuk memproses penghapusan data Pengguna.	
+ PencarianPengguna() Fungsi yang digunakan untuk memproses pencarian data Pengguna.	
+ ValidationPenggunaField() Fungsi yang digunakan untuk melakukan validasi data Pengguna yang diinputkan Pengguna.	



#### 2.2.2.4 Specific Design Class PengelolaanMember

PengelolaanMember	<<boundary>>
<pre> + PengelolaanMember() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + HapusDataMember() Fungsi yang digunakan untuk memproses penghapusan data member. + PencarianMember() Fungsi yang digunakan untuk memproses pencarian data member. </pre>	

#### 2.2.2.5 Specific Design Class PengelolaanStudio

PengelolaanStudio	<<boundary>>
<pre> + PengelolaanStudio() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + SimpanDataStudio() Fungsi yang digunakan untuk memproses penyimpanan data Studio Musik. + HapusDataStudio() Fungsi yang digunakan untuk memproses penghapusan data studio musik. + PencarianStudio() Fungsi yang digunakan untuk memproses pencarian data Studio Musik. </pre>	

#### 2.2.2.6 Specific Design Class PengelolaanBooking

PengelolaanBooking		<<boundary>>
+ PengelolaanBooking()		
Default konstruktor, digunakan untuk inisialisasi semua		
Program Studi Teknik Informatika	DPPL - MSBS	27/ 54
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

atribut dari kelas ini.

+ PembatalanBooking()

Fungsi yang digunakan untuk memproses pembatalan data booking.

+ PencarianBooking()

Fungsi yang digunakan untuk memproses pencarian data booking.

#### 2.2.2.7 Specific Design Class PengelolaanPertanyaan

PengelolaanPertanyaan	<<boundary>>
<pre>+ PengelolaanPertanyaan() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + SimpanJawabanPertanyaan() Fungsi yang digunakan untuk memproses penyimpanan jawaban pertanyaan. + PencarianPertanyaan() Fungsi yang digunakan untuk memproses pencarian data pertanyaan.</pre>	

#### 2.2.2.8 Specific Design Class Tampil Report

TampilReport	<<boundary>>
<pre>+ TampilReport() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.  + LaporanHarian() Fungsi yang digunakan untuk memproses pencetakan data pemesanan harian.</pre>	

#### 2.2.2.9 Specific Design Class Registrasi

Registrasi	<<boundary>>

Program Studi Teknik Informatika	DPPL - MSBS	28/ 54
----------------------------------	-------------	--------

+ RegistrasiMember()

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+ AddMember()

Fungsi yang digunakan untuk memproses penyimpanan data member.

+ ValidationMemberField()

Fungsi yang digunakan untuk melakukan validasi data member yang diinputkan user.

#### 2.2.2.10 Specific Design Class UbahProfil

UbahProfil	<<boundary>>
<pre>+ UbahProfil() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + EditMember() Fungsi yang digunakan untuk memproses penyimpanan perubahan profil member. + ValidationProfileField() Fungsi yang digunakan untuk melakukan validasi data member yang diinputkan member.</pre>	

#### 2.2.2.11 Specific Design Class HubungiKami

HubungiKami	<<boundary>>
<pre>+ HubungiKami() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + AddContact() Fungsi yang digunakan untuk memproses penyimpanan pertanyaan dari member kepada manajemen studio musik. + ValidationQuestionField() Fungsi yang digunakan untuk melakukan validasi data</pre>	

pertanyaan yang diinputkan member.

#### 2.2.2.12 Specific Design Class Booking

Booking	<<boundary>>
<pre>+ Booking() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + AddBooking() Fungsi yang digunakan untuk memproses penyimpanan jadwal booking dari member kepada pihak studio musik. + PembatalanPemesanan() Fungsi yang digunakan untuk memproses pembatalan jadwal booking.</pre>	

#### 2.2.2.13 Specific Design Class LoginManager

LoginManager	<<control>>
<pre>+ LoginManager() Digunakan sebagai default konstruktor dari kelas ini. + Login() Digunakan untuk melakukan proses login.</pre>	

#### 2.2.2.14 Specific Design Class PenggunaManager

PenggunaManager	<<control>>
<pre>+ PenggunaManager() Digunakan sebagai default konstruktor dari kelas ini. + SimpanPengguna() Digunakan untuk melakukan proses penyimpanan data Pengguna. + EditPengguna () Digunakan untuk melakukan proses pengeditan data Pengguna. + HapusPengguna () Digunakan untuk melakukan proses menghapus data Pengguna.</pre>	

```

+ TampilPegguna ()
Digunakan untuk melakukan proses menampilkan data
Pegguna.
+ CariPegguna ()
Digunakan untuk melakukan proses pencarian data Pegguna.
+ CekUsername()
Digunakan untuk melakukan pengecekan keunikan suatu
username.

```

#### 2.2.2.15 Specific Design Class MemberManager

MemberManager	<<control>>
<pre> + MemberManager() Digunakan sebagai default konstruktor dari kelas ini. + HapusMember() Digunakan untuk melakukan proses menghapus data member. + TampilMember() Digunakan untuk melakukan proses menampilkan data member. + CariMember() Digunakan untuk melakukan proses pencarian data member. + SimpanMember() Digunakan untuk melakukan proses penyimpanan data member. + SimpanPerubahan() Digunakan untuk melakukan proses penyimpanan data user yang telah diedit. + GetDataMember() Digunakan untuk melakukan proses mendapatkan data member tertentu. </pre>	

#### 2.2.2.16 Specific Design Class StudioManager

StudioManager	<<control>>
<pre> + StudioManager() Digunakan sebagai default konstruktor dari kelas ini. + SimpanStudio() </pre>	

Digunakan untuk melakukan proses penyimpanan data Studio.

+ EditStudio()

Digunakan untuk melakukan proses pengeditan data Studio.

+ HapusStudio()

Digunakan untuk melakukan proses menghapus data Studio.

+ TampilStudio()

Digunakan untuk melakukan proses menampilkan data Studio.

+ CariStudio()

Digunakan untuk melakukan proses pencarian data Studio.

#### 2.2.2.17 Specific Design Class BookingManager

BookingManager	<<control>>
<pre>+ BookingManager() Digunakan sebagai default konstruktor dari kelas ini. + BatalanBooking() Digunakan untuk melakukan proses pembatalan data Booking. + TampilBooking() Digunakan untuk melakukan proses menampilkan data Booking. + CariBooking() Digunakan untuk melakukan proses pencarian data Booking. + SimpanBooking() Digunakan untuk melakukan proses penyimpanan data Booking.</pre>	

#### 2.2.2.18 Specific Design Class ContactManager

ContactManager	<<control>>
<pre>+ ContactManager() Digunakan sebagai default konstruktor dari kelas ini. + SimpanJawaban() Digunakan untuk melakukan proses penyimpanan jawaban. + TampilPertanyaan() Digunakan untuk melakukan proses menampilkan data</pre>	

pertanyaan.

+ CariPertanyaan()

Digunakan untuk melakukan proses pencarian data pertanyaan.

+ SimpanPertanyaan()

Digunakan untuk melakukan proses penyimpanan pertanyaan dari user kepada manajemen studio musik.

#### 2.2.2.19 Specific Design Class ReportManager

ReportManager	<<control>>
+ ReportManager() Digunakan sebagai default konstruktor dari kelas ini.	
+ LaporanHarian() Digunakan untuk melakukan proses pencetakan data pemesanan harian.	

#### 2.2.2.20 Specific Design Class Pengguna

Pengguna	<<entity>>
- username : String Atribut untuk menyimpan data username	
- password : String Atribut untuk menyimpan data password	
- role : String Atribut untuk menyimpan data jabatan	
+ Login() Merupakan fungsi untuk melakukan login.	
+ SimpanDataPengguna() Merupakan fungsi untuk melakukan penyimpanan data Pengguna.	
+ EditDataPengguna() Merupakan fungsi untuk melakukan pengeditan data Pengguna.	
+ HapusDataPengguna() Merupakan fungsi untuk melakukan penghapusan data	

Pengguna.

+ TampilDataPengguna ()

Merupakan fungsi untuk menampilkan data Pengguna.

+ CariDataPengguna ()

Merupakan fungsi untuk melakukan pencarian data Pengguna.

+ CekUsername()

Merupakan fungsi untuk melakukan pengecekan terhadap keunikan suatu username.

#### 2.2.2.21 Specific Design Class Member

Member	<<entity>>
<p>- id_member : String Atribut untuk menyimpan data ID member</p> <p>- nama_member : string Atribut untuk menyimpan data nama member</p> <p>- alamat_member : string Atribut untuk menyimpan data alamat member</p> <p>- no_telp_member : string Atribut untuk menyimpan data nomor telepon member</p> <p>- no_hp_member : string Atribut untuk menyimpan data nomor hp member</p> <p>- email_member : String Atribut untuk menyimpan data email member</p> <p>- username_member : string Atribut untuk menyimpan data username member</p> <p>- password_member : string Atribut untuk menyimpan data password member</p>	
<p>+ SimpanDataMember() Merupakan fungsi untuk melakukan penyimpanan data member.</p> <p>+ EditDataMember() Merupakan fungsi untuk melakukan pengeditan data member.</p> <p>+ HapusDatamember()</p>	



Merupakan fungsi untuk melakukan penghapusan data member.

+ TampilDataMember()

Merupakan fungsi untuk menampilkan data member.

+ CariDataMember()

Merupakan fungsi untuk melakukan pencarian data member.

#### 2.2.2.22 Specific Design Class Studio

diskon	<<entity>>
<p>- id_studio : Integer Atribut untuk menyimpan data ID diskon</p> <p>- jenis_studio : String Atribut untuk menyimpan data jenis studio</p> <p>- harga_studio : Float Atribut untuk menyimpan data harga studio musik.</p> <p>- keterangan : string Atribut untuk menyimpan data keterangan tentang tiap ruang studio musik.</p>	
<p>+ SimpanDataStudio() Merupakan fungsi untuk melakukan penyimpanan data studio.</p> <p>+ EditDataStudio() Merupakan fungsi untuk melakukan pengeditan data studio.</p> <p>+ TampilDataStudio() Merupakan fungsi untuk menampilkan data studio.</p> <p>+ CariDataStudio() Merupakan fungsi untuk melakukan pencarian data studio.</p>	

#### 2.2.2.23 Specific Design Class Booking

Booking	<<entity>>
<p>- id_booking : Integer Atribut untuk menyimpan data ID booking</p> <p>- tgl_booking : Datetime Atribut untuk menyimpan data tanggal booking</p> <p>- status_booking : string</p>	

<p>Atribut untuk menyimpan data status booking</p> <p>- jam_mulai : string</p> <p>Atribut untuk menyimpan data jam mulai</p> <p>- jam_selesai : string</p> <p>Atribut untuk menyimpan data jam selesai</p> <p>- total_biaya : Float</p> <p>Atribut untuk menyimpan data total biaya dari hasil perkalian harga sewa studio dengan lama sewa studio.</p>
<p>+ SimpanDataBooking()</p> <p>Merupakan fungsi untuk melakukan penyimpanan data Booking.</p> <p>+ TampilDataBooking()</p> <p>Merupakan fungsi untuk menampilkan data booking.</p> <p>+ CariDataBooking ()</p> <p>Merupakan fungsi untuk melakukan pencarian data booking.</p> <p>+ BatalkanBooking()</p> <p>Merupakan fungsi untuk melakukan pembatalan proses booking.</p> <p>+ CetakReportPemesanan()</p> <p>Merupakan fungsi untuk melakukan pencetakan laporan pemesanan.</p>

#### 2.2.2.24 Specific Design Class Contact

Contact	<<entity>>
<p>- id_contact : Integer</p> <p>Atribut untuk menyimpan data ID contact</p> <p>- tgl_pertanyaan : Datetime</p> <p>Atribut untuk menyimpan data tanggal pengiriman pertanyaan</p> <p>- pertanyaan : String</p> <p>Atribut untuk menyimpan data isi dari pertanyaan</p> <p>- jawaban : String</p> <p>Atribut untuk menyimpan data isi dari jawaban</p> <p>+ SimpanDataPertanyaan()</p>	

Program Studi Teknik Informatika	DPPL - MSBS	36/ 54
----------------------------------	-------------	--------

Merupakan fungsi untuk melakukan penyimpanan data pertanyaan.

+ JawabPertanyaan()

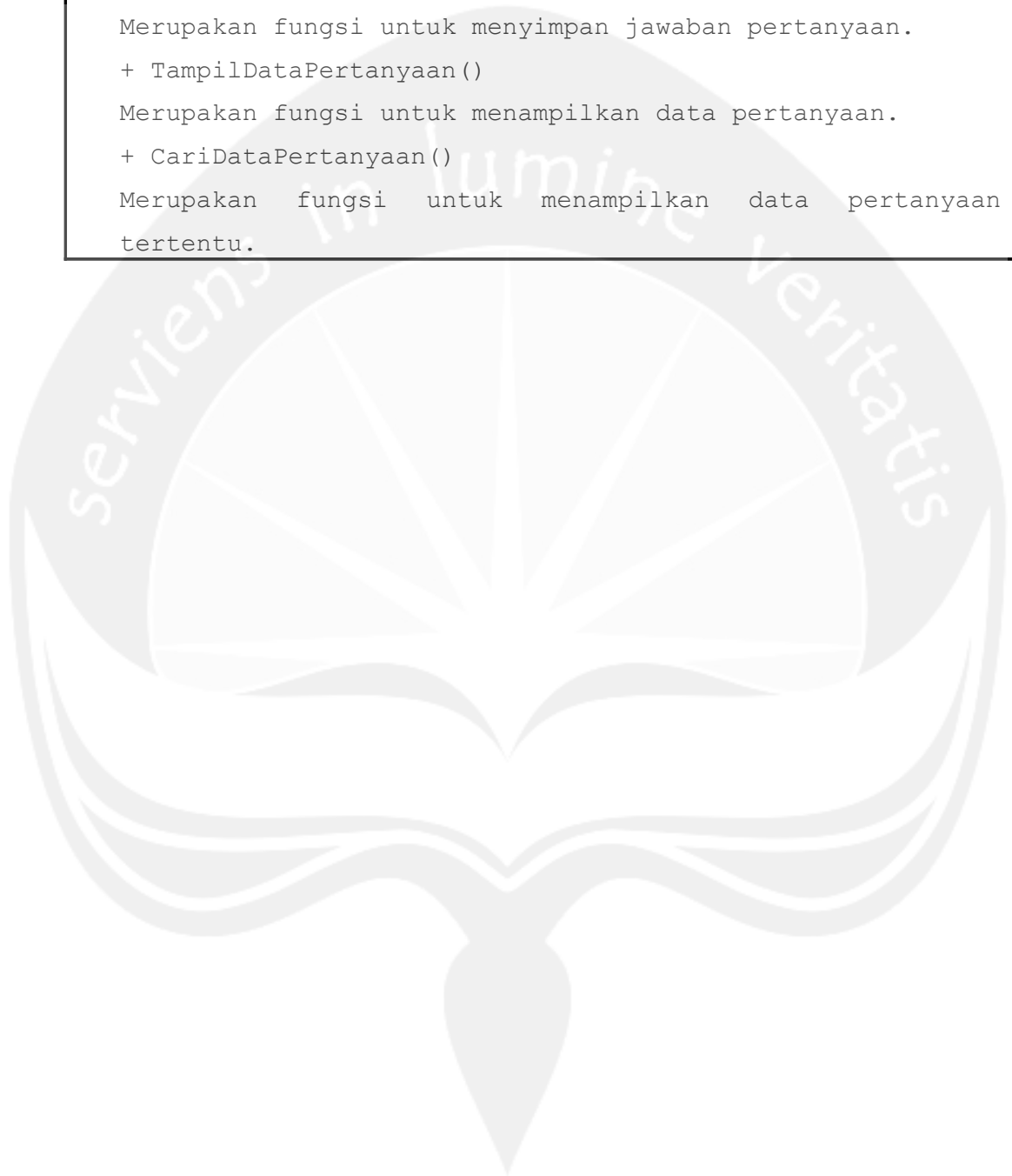
Merupakan fungsi untuk menyimpan jawaban pertanyaan.

+ TampilDataPertanyaan()

Merupakan fungsi untuk menampilkan data pertanyaan.

+ CariDataPertanyaan()

Merupakan fungsi untuk menampilkan data pertanyaan tertentu.



### 3 Perancangan Data

#### 3.1 Dekomposisi Data

##### 3.1.1 Deskripsi Entitas Data Pengguna

Nama	Tipe	Panjang	Keterangan
<b>id_pengguna</b>	<b>Integer</b>	<b>-</b>	<b>ID Pengguna, Primary Key</b>
username	Character	10	Username user
password	Character	10	Password user
roles	Character	20	Jabatan user

##### 3.1.2 Deskripsi Entitas Data Studio

Nama	Tipe	Panjang	Keterangan
<b>id_studio</b>	<b>Integer</b>	<b>-</b>	<b>ID studio musik, Primary key</b>
Jenis_studio	Character	20	Jenis studio
Harga_studio	Float	-	Harga studio
Keterangan	Character	100	Keterangan Studio

##### 3.1.3 Deskripsi Entitas Data Member

Nama	Tipe	Panjang	Keterangan
<b>id_member</b>	<b>Character</b>	<b>10</b>	<b>ID barang, Primary key</b>
Nama_member	Integer	-	Nama member
Alamat_member	Integer	-	Alamat member
no_telp_member	Float	-	Nomor telepon member
no_hp_member	Float	-	Nomor Handphone member
email_member	Character	20	Email member
Username_member	Character	10	Username member
Password_member	Character	10	Password member

##### 3.1.4 Deskripsi Entitas Data Booking

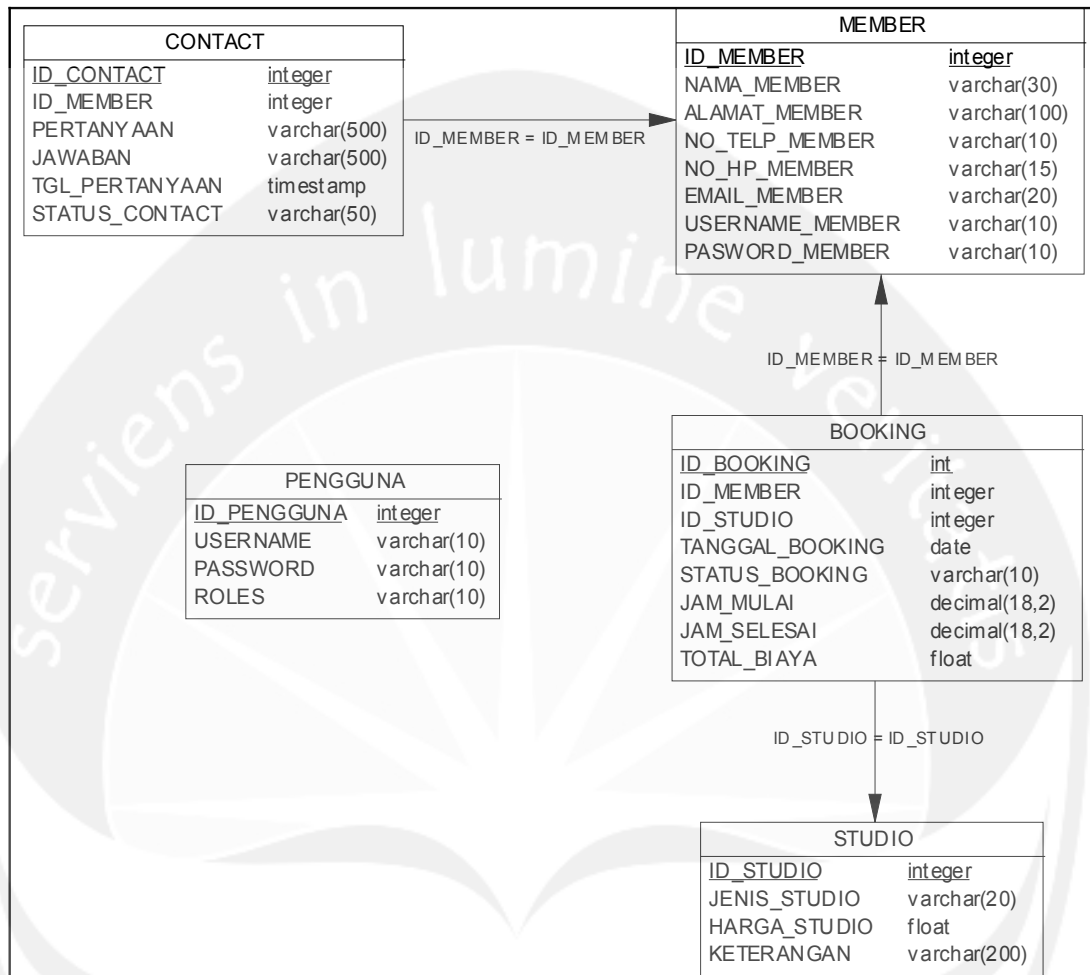
Nama	Tipe	Panjang	Keterangan
------	------	---------	------------

<b>id_booking</b>	<b>Integer</b>	-	<b>ID booking, Primary key</b>
id_studio	<b>Integer</b>	-	ID Studio, foreign key
id_member	<b>Integer</b>	-	ID member, foreign key
tanggal_booking	DateTime	-	Tanggal booking
status_booking	Character	10	Status dari booking
Jam_mulai	Decimal	18,2	Jam mulai booking
Jam_selesai	Decimal	18,2	Jam selesai booking
Total_biaya	Float	-	Total biaya booking

### 3.1.5 Deskripsi Entitas Data Contact

Nama	Tipe	Panjang	Keterangan
<b>id_contact</b>	<b>Integer</b>	-	<b>ID contact, Primary key</b>
id_member	Integer	-	Id member, foreign key
pertanyaan	Character	200	Pertanyaan, kritik, atau saran yang diajukan pengirim
jawaban	Character	200	Jawaban atau tanggapan perusahaan
tgl_pertanyaan	DateTime	-	Tanggal pertanyaan dikirimkan

### 3.2 PDM



Gambar 3.1 PDM

## 4 Perancangan Antarmuka

### 4.1. Login

LOGO

TELP  
EMAIL

HOME | TENTANG KAMI | GALERI FOTO | SERVIS | JADWAL | HUBUNGI KAMI | LOGIN

LOGIN

username  
password

login

Daftar Member? Daftar

Copyright © EHA Studio Ragus Perwira Laksmiana - All right reserved

Gambar 4.1 Perancangan Antarmuka - Login Member

LOGO

TELP  
EMAIL

HOME | TENTANG KAMI | GALERI FOTO | SERVIS | JADWAL | HUBUNGI KAMI | LOGIN

LOGIN

username  
password

login

Daftar Member? Daftar

Copyright © EHA Studio Ragus Perwira Laksmiana - All right reserved

Gambar 4.2 Perancangan Antarmuka - Login Pengelolaan

Antarmuka pertama yang akan ditemui user adalah antarmuka login. Terdapat 2 antarmuka login yang tersedia, yaitu antarmuka login untuk member (gambar 4.1) dan antarmuka login untuk pengelolaan data (gambar 4.2). Kedua antarmuka tersebut memiliki desain yang sama, yaitu memiliki 2 buah *textfield*, untuk *username* dan *password*, memiliki 1 buah *button* untuk login. Khusus untuk login member, terdapat *link button* lain dengan teks 'Daftar Member' yang digunakan untuk registrasi member.

Sistem akan melakukan pengecekan data *username* dan *password* yang dimasukkan user setelah user menekan tombol login. Apabila data tidak ditemukan, maka akan muncul peringatan bahwa login gagal dilakukan.

#### 4.2. Pengelolaan Data Pengguna

Gambar 4.3 Perancangan Antarmuka - Pengelolaan Data Pengguna

Antarmuka berikutnya adalah pengelolaan data pengguna (gambar 4.3). Untuk bagian input data terdapat 3 buah textfield, 1 buah combobox, serta 1 button. 3 buah textfield pertama masing-masing untuk menginputkan username, password, serta re-password untuk dicocokkan dengan inputan password. Lalu juga terdapat 1 combobox untuk memilih Role user dimana terdapat 3 pilihan yaitu administrator, operator, dan manager. Kemudian terdapat 1 buah button untuk simpan data.

Selanjutnya pada bagian menampilkan, mencari, dan menghapus data. Di sini terdapat sebuah *datagridview* untuk menampilkan data user. Lalu terdapat juga 1 buah textfield



yang digunakan untuk inputan pencarian data berdasarkan username, serta 2 buah *button*, masing-masing untuk search data secara khusus serta *button* display all untuk menampilkan semua data. Untuk melakukan pengeditan atau penghapusan data, user harus memilih salah satu data dalam *datagrid*.

#### 4.3. Pengelolaan Data Member

Gambar 4.4 Perancangan Antarmuka - Pengelolaan Data Member

Pengelolaan member akan memiliki fungsi untuk mencari, menampilkan, dan menghapus data saja. Oleh karena itu hanya terdapat sebuah antarmuka yang akan dibuat. Pada antarmuka pengelolaan member akan terdapat sebuah *datagrid* yang akan digunakan untuk menampilkan data member. Untuk melakukan delete data telah disediakan menu delete di dalam *datagrid*. Selanjutnya sistem akan melakukan proses penghapusan data.

Untuk pencarian data, terdapat sebuah *textfield* untuk menginputkan nama member yang akan dicari. Terdapat 2 buah *button* yaitu *button* search untuk melakukan pencarian data secara khusus berdasarkan inputan, serta *button* display all untuk menampilkan semua data member.

#### 4.4. Pengelolaan Data Pertanyaan

LOGO

TELP  
EMAIL

Home Kelola Booking Kelola Pertanyaan Logout

PENGELOLAAN DATA PERTANYAAN

Display Data

Cari berdasarkan status

search Display all

Jawab Pertanyaan

Kirim

Copyright © EHA Studio-Bagus Penwiria Lakomana - All right reserved

Gambar 4.5 Perancangan Antarmuka - Pengelolaan Data Pertanyaan

Pengelolaan data pertanyaan memiliki sebuah antarmuka utama (gambar 4.6). Terdapat sebuah *combobox* untuk mencari pertanyaan berdasarkan status yaitu "terjawab" dan "belum terjawab". Kemudian terdapat 2 button masing-masing *search* dan *display all*. Terdapat sebuah *datagridview* untuk menampilkan data pertanyaan dari member. Pada *datagrid* tersebut juga terdapat menu *edit* serta *delete*. Menu *edit* digunakan untuk melakukan jawab pertanyaan. Untuk menjawab pertanyaan disediakan 1 buah *textfield* serta 1 buah button "kirim" untuk melakukan penyimpanan data jawaban.

#### 4.5. Pengelolaan Data Booking

LOGO

TELP  
EMAIL

Home Kelola Booking Kelola Pertanyaan Logout

PENGELOLAAN DATA BOOKING

Display Data

Cari berdasarkan tanggal booking

oktober

search Display all

Copyright © EHA Studio Bagus Perwira Laksmiana - All right reserved

○○○○

Gambar 4.6 Perancangan Antarmuka - Pengelolaan Data Booking

Pengelolaan data booking memiliki sebuah antarmuka utama (gambar 4.6). Terdapat sebuah kalender untuk pemilihan tanggal booking karena untuk pencarian data booking berdasarkan tanggal booking. Kemudian terdapat 2 button. Button tersebut masing-masing button search dan button display all. Button search digunakan untuk melakukan pencarian data berdasarkan criteria tertentu, kriteria di sini yang dimaksud adalah pencarian data berdasarkan tanggal booking. Sedangkan button display all digunakan untuk menampilkan semua data booking yang ada pada database tanpa

criteria tertentu. Untuk menampilkan data disediakan sebuah gridview. Gridview disediakan untuk tempat menampilkan semua data yang ada pada database. Pada gridview terdapat menu delete. Menu delete hanya digunakan untuk menghapus data yang tidak digunakan atau batal booking.

#### 4.6. Pengelolaan Data Studio

LOGO

TELP  
EMAIL

Home Kelola Studio Tampil Laporan Logout

PENGLOLAAN DATA STUDIO

Jenis Studio  
Harga Studio  
Keterangan

Save  
Display Data

Search by Jenis Studio

Search

Copyright © EHA Studio Bagus Perwira Laksmiana - All right reserved

Gambar 4.7 Perancangan Antarmuka - Pengelolaan Data Studio

Pengelolaan data studio dilakukan oleh seorang manager. Pengelolaan ini memiliki sebuah antarmuka utama (gambar 4.7). Terdapat 3 buah *textfield* untuk menginputkan

data. Masing-masing yaitu *textfield* untuk menginputkan data jenis studio, *textfield* untuk menginputkan data harga studio, dan *textfield* untuk menginputkan data keterangan. Kemudian dibawah ketiga *textfield* tersebut terdapat sebuah button save untuk menyimpan ketiga buah data yang sudah diketikkan. Selain itu terdapat pula 1 buah *combobox* untuk melakukan pencarian berdasarkan jenis studio dan 1 buah button search untuk melakukan pencarian berdasarkan criteria tertentu. Kriteria di sini yang dimaksud adalah pencarian data berdasarkan jenis studio. Kemudian disediakan sebuah gridview untuk menampilkan data studio. Pada gridview terdapat 2 buah menu yaitu edit dan delete. Menu edit digunakan untuk melakukan update data studio berdasarkan *id\_studio* yang akan diedit. Sedangkan menu delete digunakan untuk melakukan hapus data berdasarkan *id\_studio* yang dipilih.

#### 4.7. Tampil Laporan Pemesanan Harian

LOGO

TELP  
EMAIL

Home Kelda Studio Tampil Laporan Logout

TAMPIL REPORT HARIAN

Pilih Tanggal

Pilih

Copyright © EHA Studio-Bagus Perwira Laksmiana - All right reserved

Gambar 4.8 Perancangan Antarmuka -  
Tampil report Harian

Untuk tampil laporan pemesanan harian, hanya terdapat 1 buah kalender untuk melakukan pemilihan tanggal yang kemudian akan ditampilkan pada crystal report. Untuk mendukung kalender tersebut disediakan sebuah button "Pilih" untuk melakukan pemilihan tanggal berdasarkan kriteria

tertentu. Kriteria di sini yang dimaksud adalah tanggal yang dipilih dari kalender. Kemudian terdapat sebuah *crystal report* yang digunakan untuk menampilkan data pemesanan harian berdasarkan tanggal yang sudah dipilih. Crystal report akan menampilkan semua data booking yang ada pada database.

#### 4.8. Registrasi Member

Gambar 4.9 Perancangan Antarmuka - Registrasi Member

Antarmuka registrasi member terdiri dari 7 *textfield* masing-masing untuk menginputkan data nama, alamat, nomor telepon, nomor HP, email, username, serta password dan 1 buah *button* "Submit". *Textfield* disediakan untuk user memasukkan data member yang akan disimpan. Setelah data dimasukkan, maka user harus menekan *button* 'Submit' untuk melakukan proses penyimpanan data member. *Username* yang

dimasukkan harus unik. Jika *username* sudah ada, maka akan muncul peringatan bahwa *username* sudah ada dalam *database*. Dan jika penyimpanan data registrasi member berhasil dilakukan, maka akan muncul pemberitahuan bahwa registrasi berhasil dilakukan.

#### 4.9. Ubah Profil Member

LOGO

TELP  
EMAIL

HOME UBAH PROFIL SERVIS JADWAL BOOKING HUBUNGI KAMI LOGIN

UBAH PROFIL MEMBER

Nama

Alamat

Nomor Telepon

Nomor HP

Email

Username

Password

Simpan

HOME | TENTANG KAMI | GALERI FOTO | SERVIS | JADWAL | HUBUNGI KAMI | LOGIN

Copyright © EHA Studio-Bagus Perwira Laksmiana – All right reserved

Gambar 4.10 Perancangan Antarmuka - Ubah Profil Member

Antarmuka untuk mengubah data member dapat ditampilkan jika member sudah melakukan proses login. Dalam antarmuka ini terdapat 7 buah *textfield* dan 1 buah *button* "simpan". *Textfield* tersebut digunakan untuk user mengedit data member. Setelah data dimasukkan, maka user dapat menekan



*button* 'Simpan' untuk melakukan proses penyimpanan. Jika penyimpanan berhasil dilakukan, akan muncul pemberitahuan bahwa data berhasil diubah.

#### **4.10. Pengiriman Pertanyaan**

Antarmuka untuk mengirimkan pertanyaan, kritik, atau saran kepada manajemen studio musik akan dapat dilakukan melalui halaman 'Hubungi Kami'. Pada antarmuka ini akan disediakan 1 buah *textfield* untuk memasukkan isi pertanyaan. Setelah itu user dapat menekan *button* 'Kirim' yang akan disediakan untuk melakukan proses penyimpanan data pertanyaan. Teks yang disediakan di antarmuka ini akan menampilkan sekilas tentang alamat, nomor telepon, dan email dari studio musik.

LOGO

TELP  
EMAIL

HOME

UBAH PROFIL

SERVIS

JADWAL

BOOKING

HUBUNGI KAMI

LOGIN

HUBUNGI KAMI

ADDRESS

EHA MUSIC STUDIO  
Jalan Palagan no. 502,  
Sleman, Jogjakarta,  
Indonesia

TELEPHONE

0274-517377

EMAIL

eha@studios.co.id

Masukkan pertanyaan/ kritik/ saran:

Kirim

Clear

HOME | TENTANG KAMI | GALERI FOTO | SERVIS | JADWAL | HUBUNGI KAMI | LOGIN

Copyright © EHA Studio-Bagus Perwira Laksmana – All right reserved

Gambar 4.11 Perancangan Antarmuka – Pengiriman  
Pertanyaan

#### 4.11. Booking Studio Musik

Fungsi booking studio musik akan memiliki 1 buah antarmuka utama seperti pada gambar 4.12. Pada antarmuka tersebut terdapat sebuah *textfield* dan 3 buah *combobox* yang digunakan untuk memasukkan data booking studio yang dikehendaki. Combobox pertama digunakan untuk memilih data jenis studio, kemudian *textfield* pilih tanggal, serta 2 combobox yang masing-masing untuk memilih jam mulai dan jam selesai. Setelah data sudah dimasukkan lalu disediakan

button 'submit' untuk melakukan penyimpanan data booking. Setelah itu akan tampil sebuah form yang berisi tentang data-data yang barusan diinputkan oleh member.

Gambar 4.12 Perancangan Antarmuka – Pengecekan Data Booking

#### 4.12. Lihat Jadwal Studio Musik

Fungsi lihat jadwal studio musik akan memiliki 1 buah antarmuka utama seperti pada gambar 4.13. Pada antarmuka tersebut terdapat sebuah kalender dan 1 buah button yang digunakan untuk memasukkan data tanggal yang dikehendaki. Setelah data tanggal sudah dimasukkan lalu disediakan button 'cek jadwal' untuk menampilkan data jadwal. Setelah itu

akan tampil sebuah form yang berisi tentang data-data yang barusan diinputkan oleh member.

LOGO

TELP  
EMAIL

HOME UBAH PROFIL SERVIS JADWAL BOOKING HUBUNGI KAMI LOGIN

JADWAL

Pilih tanggal

Cek jadwal

HOME | TENTANG KAMI | GALERI FOTO | SERVIS | JADWAL | HUBUNGI KAMI | LOGIN

Copyright © EHA Studio Bagus Perwira Laksmiana - All right reserved

Gambar 4.12 Perancangan Antarmuka - Pengecekan Data Booking